Probability
for Information Technology Applications

Changho Suh[1]

June 8, 2021

[1] Changho Suh is an Associate Professor in the School of Electrical Engineering at Korea Advanced Institute of Science and Technology, South Korea (Email: chsuh@kaist.ac.kr).

EE210 Probability and Introductory Random Processes March 2, 2021
KAIST, Spring 2021 Changho Suh (chsuh@kaist.ac.kr)

Lecture 1: Logistics and Overview

About instructor

Welcome to EE210: Probability and introductory random processes! My name is Changho Suh, an instructor of this course. A brief introduction of myself. A long time ago, I was one of the students in KAIST like you. I spent six years at KAIST to obtain the Bachelor and Master degrees all from Electrical Engineering in 2000 and 2002, respectively. In 2002, I joined Samsung electronics where I worked on the design of wireless communication systems like 4G/5G. Spending four and a half years, I left industry, joining UC Berkeley where I obtained the PhD degree in 2011. Spending around one year at MIT as a postdoc, I came back to KAIST in 2012. My research interests include information theory and machine learning which of course have something to do with the concepts of probability and random processes that I am going to cover in this course.

Today's lecture

Today we will cover two very basic stuffs. The first is logistics of this course: details as to how the course is organized and will proceed. The second thing to cover is a brief overview to this course. Specifically I am going to first explain the basic concept of probability and why it is of interest in the department of Electrical Engineering. I will then argue that probability serves as an essential tool in a wide range of applications from various fields including Electrical Engineering. Lastly I will provide you with specific topics that we will cover throughout this course.

My contact information, office hours and TAs' information

See syllabus uploaded on the course website. One special note: if you cannot make it neither for my office hours nor for TAs' ones, please send me or TAs an email to make an appointment in different time slots. Office hours are basically in person, but zoom calls may be open upon request.

Prerequisite

A prerequisite for this course is to have some mathematical maturity and exposure to programming. High school level knowledge about the concept of probability suffices, as long as you were in the science division ("ik-gwa"). Some knowledge on random processes helps indeed although it is not mandatory. For programming, we will use Python, so basic familiarity with Python helps a lot. Even though you never learned about Python, it may be okay as long as you are familiar with the programming concept. We will offer a Python tutorial so that you can learn about the language by yourself. It will be issued right before Python is firstly used. If you still think that you lack these prerequisites, please consult with me so that I can help you as much as possible.

Course website

We have a course website on klms system. You can simply login with your portal ID. If you want to only sit in this course (or cannot register the course for some reason), please let me or head TA (Moonseok Choi) know your email address. Upon request, we are willing to distribute course materials to the email address that you sent us.

Text

There is no required textbook for this course. Instead I am going to provide you with two materials which I believe enough: (i) lecture slides (LSs for short) which I will use during lectures; (ii) course notes (CNs for short) like the one that you are now reading. Most times, these materials will be posted at night on a day before class, but sometimes course notes may be uploaded after lectures. These materials are almost self-contained, and cover the entire contents that will show up in homeworks and exams. So if you want to make a minimal effort to this course, these materials may suffice to pass the course with a reasonable grade. The recording of the livestream, which we call Lecture Videos (LVs), will be uploaded on youtube at the end of the corresponding week. TA0 (Youngsu Jang) will take care of this, and inform you of the youtube link once uploaded. As supplement, you can also consult with a very easy-to-follow course note at MIT (already uploaded on the course website):

> Bertsekas and Tsitsiklis, "Introduction to Probability," Course Notes @ MIT.

For those who have enough energy, passion and time, I recommend you to consult with another reference: Jean Walrand, "Probability in Electrical Engineering and Computer Science: An Application-driven Course," Amazon, 2014. Sometimes, I will make some homework problems from these references. If so, I will let you know and upload a soft copy of the relevant part of the materials.

Problem sets

There will be weekly or bi-weekly problem sets. So there would be around seven problem sets in total. Each homework will be graded by a subset of TAs only among TA1 $\sim$ TA6 (excluding head TA and TA0), and a responsible TA will be assigned to each problem (this information will be indicated in the problem set as well). If you have some question about a problem, I strongly recommend you to first ask the responsible TA and then the head TA (if needed), potentially ccing me. The submission is via klms – detailed instruction will be announced by the head TA around when the first problem set is issued. Solutions will be usually available at the end of the due date. This means that in principle, we do not accept any late submission. We encourage you to cooperate with each other in solving the problem sets. However, you should write down (and/or type via any edit tools such as latex and docs) your own solutions. You are welcome to flag confusing topics in the problem sets; this will not lower your grade. Some problems require programming and simulations in Python. To this end, we will be using Jupyter notebook: https://jupyter.org/. Please refer to the installation guide if needed: https://jupyter.readthedocs.io/en/latest/install.html.

Exams

As usual, there will be two exams: midterm and final. We follow the schedule that our institution assigns by default: Tuesday 9:00–11:30 am on the exam week for this course. Please let us know if someone cannot make it for the schedule. Upon a reasonable rationale, we can change the schedule or can offer a chance for you to take an exam in a different time slot that we will organize individually.

Three things to notice. First, an exam runs live online via Zoom. We are supposed to host 8 individual meetings, each being proctored by one responsible TA. For exams, you should be prepared with: (i) a video camera (smartphone camera suffices); (ii) a quiet place; (iii) a printer;

(iv) a scanner (smartphone app scanner suffices). An exam starts from 9 am and ends at 11:30 am sharp. Exam sheets would be uploaded at 8:30 am on klms, so you would have around 30 minutes for printing. You should turn in your solution to a corresponding TA until noon (30 minutes for scanning). You must turn on your camera from 9 am to your submission time (including your scanning time).

Second, for both exams, you are allowed to use one cheating sheet, A4-sized and double-sided. So it is a kind of semi-closed-book exam. It can be either typed or hand-written. Basically you can do whatever you want yet within one sheet of paper.

Lastly, for your convenience, we will provide an instruction note for each exam, which contains detailed guidelines as to how to prepare for the exam. Such information includes: (1) how many problems are in the exam; (2) what types of problems are dealt with in what contexts; (3) the best way to prepare for such problems.

Course grade

Here is a rule for the course grade that you are mostly interested in perhaps. The grade will be decided based on four factors: problem sets (22%); midterm (32%); final (40%); and participation (6%). Here the participation means any type of interaction with me: attendance, in-class participation, questions, discussion, email exchange, to name a few.

Overview

Now let's move onto the second part. Here is information for reading materials: "CN01". In this part, I will explain the basic concept of probability of our main interest and its role in the context of various applications. I will then list up specific topics that we will learn about throughout the course.

Probability and uncertainty

Let's start from the beginning. What is probability? The not-very-rigorous yet very intuitive definition is the following. Probability is defined as the fraction of the occurrence of an interested event over the total number of possible cases. To get a concrete feel as to what it means, let's think about rolling a dice. Suppose that an interested event is getting a particular number, say 1, as a result of rolling. Since there are six possible cases for the result, from 1 to 6, the probability simply reads $\frac{1}{6}$. Now then why do we care about probability particularly within the department of Electrical Engineering that seems to have nothing to do with rolling a dice? This is because life is full of *uncertainty.* There are tons of scenarios in the real world (including many applications in Electrical Engineering) wherein one cannot predict the future with 100%, and hence one can talk about future occurrence only in a *probabilistic* manner.

Many applications

In old days, the theory of probability has been developed mainly for the purpose of earning some money from gambling. People wanted to figure out probability-theory-based best strategies for card games, dice, roulette wheels. But today it serves as an essential tool in a widening array of applications from various fields. For instance, it is instrumental in designing *best systems* for communication, internet and control. It also helps to build up key methodologies for artificial intelligence and machine learning, called *algorithms* in the field. Often times, it plays a crucial role in understanding many important theories and principles that arise in science. Many other applications include: cloud storage, peer-to-peer file sharing, speech recognition, ranking of webpages, network multiplexing, GPS (positioning), DNA sequencing, etc. Among many of such applications, let's focus on the following three killer applications to explain the role of probability in depth: (1) communication; (2) machine learning; (3) speech recognition.

Application #1: Communication

Communication is the transfer of information (often digital information, a sequence of 0/1 binary digits, simply called bits) from one end (called the transmitter) to the other end (called the receiver), over a physical medium (like an air) between the two ends. The physical medium is so called the *channel*. See Fig. **??**.

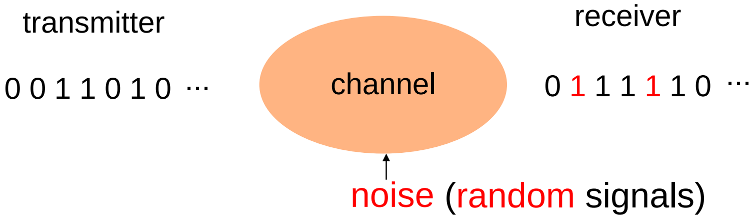


Figure 1: Communication is the transfer of information from the transmitter to the receiver over a channel. The channel introduces *random noise signals* which can then be modeled as a key concept in probability: a *sequence of random variables* (also called a *random process*).

Here the channel is the one that relates the concept of probability to communication. If you think about how the channel behaves, then you can easily see why. First of all, the channel is a sort of system (in other words, a function) which takes a transmitted signal as an input and a received signal as an output. Here one big problem arises in the system. The problem is that it is not a *deterministic* function. If the channel is deterministic and one-to-one mapping, then one can easily reconstruct the input from the output. So there is no problem in transferring information from the transmitter to the receiver. However, the channel is not deterministic in reality. It is actually a *random* function. In other words, there is a random entity (also known as *noise* in the field) added into the system. In typical communication systems, the noise is additive: a received signal is the sum of a transmitted signal and the noise. In general, we have no idea of the noise. It turns out the random noise signals can be mathematically modeled as the one that you may often hear of during high school days: a *sequence of random variables*, also simply called a *random process*. *Random variable* and *random processes* are indeed key concepts in probability. Here we see its relevance to probability.

Actually it is more than that if we think about the goal of communication. The goal of communication is to reconstruct the original transmitted signals ($0011010\cdots$ in the example in Fig. **??**) from the corrupted received signals ($0111110\cdots$ in the example). So one natural desire in this context is to figure out the best way of reconstruction. It turns out the best way builds upon one key principle in probability: Maximum A Posteriori (MAP) estimation. Many of you guys may not hear of the MAP estimation. Don't worry. That is exactly the reason that you are taking this course. We will study the MAP principle in depth from this course.

Application #2: Machine learning

Probability is also essential in understanding a very trending field nowadays that most of you guys are sort of forced to be interested in: Machine learning. Machine learning is a methodology for training a machine so that the trained machine can perform like human beings. See Fig. **??**. Here one key feature of machine learning is that we use *data* in the process of training a machine.

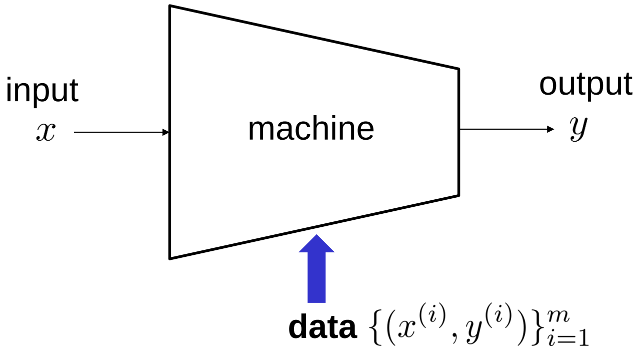


Figure 2: Machine learning is a methodology for training a machine so that it can perform as much as like human beings. For certain yet interested scenarios, the best training methodology builds upon one key principle in probability: Maximum Likelihood estimation.

The data often refers to input-output paired samples, denoted by:

$$\{(x^{(i)}, y^{(i)})\}_{i=1}^m, \tag{1}$$

where $(x^{(i)}, y^{(i)})$ indicates the ith input-output sample (or called a training sample or an example) and m denotes the number of samples.

In fact, probability is related to a training methodology. It turns out under certain scenarios that the best way of training a machine hinges upon another key principle in probability: Maximum Likelihood (ML) estimation. Most of you guys may not hear of the ML estimation. Again, don't worry. This is why you are here. We will explore in depth this important principle.

Application #3: Speech recognition

Probability is also known to be instrumental in designing a popular system prevalent in our daily life: Speech recognition. Siri in the iphone is one such example. Amazon Alexa is another. Actually, you can see its relevance to probability if you think about the goal of speech recognition. The goal of speech recognition is to transform voice signals (comprising spoken words picked up from microphone) into a written command, which can then be represented in the form of a *text* without losing the meaning of the spoken words. See Fig. **??**. Here the key observation is

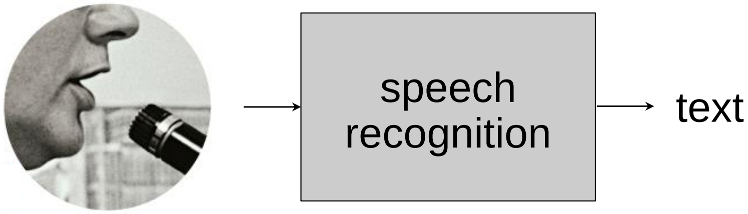


Figure 3: The goal of speech recognition is to transform voice signals into a written command. Here voice characteristics (also called pronunciations) vary over different speakers. Hence, the voice signals with uncertainty can be modeled as a random process. Also it is well known that the best way for designing a speech recognition system is based on a key principle in probability: Maximum A Posteriori (MAP) estimation.

that voice characteristics vary over different speakers, which in turn incurs uncertainty on voice signals. So one can model the voice signals with uncertainty as a *random process.* Another relevance comes from a way of designing speech recognition systems. It turns out that the problem of speech recognition can be cast into the communication problem that we discussed in Fig. **??**, so the best way of designing speech recognition is again based on the MAP estimation.

Contents to be covered

From the earlier killer applications, we see several key concepts and principles: (i) random variables; (ii) random processes; (iii) MAP estimation; (iv) ML estimation. Actually these are the main contents that I would like to explain to you from this course. A little bit exaggeratedly speaking, with a deep understanding only on these, you can do almost whatever you want at least within the field of Electrical Engineering. Unfortunately, however, understanding these requires some knowledge on more basic concepts of probability. That's why I structured the course with the following three parts.

Course outline

In Part I, we will study such basic concepts of probability. These are actually the ones that you may hear of from your high school math: sample space, events, conditional probability, independence, total probability law, random variable, probability mass function, density and expectation. In Part II, we will then move onto the key principles: MAP and ML estimation. It turns out that the derivation of the MAP and ML estimation requires two more key principles: (a) Law of Large Number; (b) Central Limit Theorem. We will also study these in Part II. In addition, we will cover popular inequality techniques that are useful in proving the theorems concerning the key principles: Markov's inequality and Chebyshev's inequality. In the last third part, we will explore the three killer applications (communication, machine learning, speech recognition) and will emphasize the role of the key concepts again yet more convincingly using the knowledge acquired in Parts I and II.

EE210 Probability and Introductory Random Processes — March 4, 2021

KAIST, Spring 2021 — Changho Suh (chsuh@kaist.ac.kr)

Lecture 2: Sample space and events

Recap

Last time, we introduced the non-rigorous yet intuitive definition of probability: the fraction of the occurrences of an interested event over all the possible outcomes of an experiment. We then emphasized the role of probability while explaining its role in the context of the following three killer applications: (i) communication; (ii) machine learning; and (iii) speech recognition. In particular, we put a special emphasis on two key concepts and two key principles. The two concepts are: (i) random variables; (ii) random processes. The two principles are: (i) MAP (Maximum A Posteriori) estimation; (ii) ML (Maximum Likelihood) estimation.

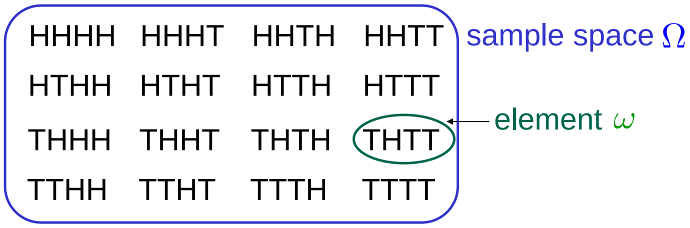
At the end of the last lecture, however, I claimed that understanding these requires some knowledge on more basic concepts of probability like sample space, events, conditional probability, independence and total probability law. During several upcoming lectures, we will explore them in depth.

Today's lecture

Today we will learn about the first two: sample space and events. Specifically what we are going to cover are four folded. First we will introduce the definition of sample space. We will then study its relevant concept, called the probability (or probabilistic) model, which builds upon the sample space and also play a role in bridging the sample space to events. Next we will investigate the definition of an event, as well as study how to compute the probability of an interested event. Finally we will exercise ourselves on the learned concepts via several examples: five easy and one non-trivial examples.

Sample space

The sample space is defined as the set of all possible outcomes of an experiment. To get a concrete feel, let's think about a simple experiment where we toss a coin four times. In this case, what are the possible outcomes? They would read: HHHH, HHHT, HHTH, HHTT, all the way up to, TTTH, TTTT. See Fig. 1. So the sample space, usually denoted by Ω, would be: $\Omega = \{HHHT, HHHT, HHTH, HHTT, \ldots, TTTH, TTTT\}$. Here an element in the set,



HHHH	HHHT	HHTH	HHTT
HTHH	HTHT	HTTH	HTTT
THHH	THHT	THTH	THTT
TTHH	TTHT	TTTH	TTTT

Figure 1: Sample space for an experiment of tossing a coin four times. The sample space is defined as the *set* of all possible outcomes, and is conventionally denoted by Ω. Here H and T stand for "Head" and "Tail" respectively. An element in the set indicates each possible outcome and is conventionally denoted by a small letter ω.

usually denoted by $\omega \in \Omega$, indicates each outcome. The sample space is connected to the concept of events via another relevant concept: *probability model*.

Probability model

A probability (or probabilistic) model is a mathematical description that consists of the following two entities. One is the sample space Ω. The second is a so called *probability assignment*, denoted by $\mathbb{P}(\omega)$, for each $\omega \in \Omega$. Here the probability assignment (also called the *probability distribution*) should respect the non-negativity and sum-up-to-one properties:

$$0 \leq \mathbb{P}(\omega) \leq 1 \qquad \text{for all } \omega \in \Omega; \tag{1}$$

$$\sum_{\omega \in \Omega} \mathbb{P}(\omega) = 1. \tag{2}$$

The easiest way to assign probabilities to outcomes is to give all of them the same probability (as we saw earlier in the coin tossing example; see Fig. 1): $\mathbb{P}(\omega) = \frac{1}{|\Omega|}\ \ \forall \omega \in \Omega$. Here the symbol "$\forall$" means "for all" and $|\Omega|$ indicates the number of all the elements in Ω, simply called the *cardinality* of Ω. This distribution is known as a *uniform distribution*. Of course, not all the probability distributions are uniform. We will also see examples of non-uniform distributions soon. Notice that while there are many outcomes in the sample space, the experiment results in exactly only one of these outcomes. However, we do not know in advance the future outcome, so lies the randomness here. The only thing that we can do is to talk about such occurrence in a *probabilistic* manner via proper modeling of the probability distribution $\mathbb{P}(\omega)$.

Events

In the context of probability theory, an event means a certain interested occurrence in an experiment. To get a concrete feel as to what it means, let's recall the earlier coin-tossing example; again see Fig. 1. Suppose we are interested in an occurrence where we get "two heads" as a result of four-times coin tossing. Then, the corresponding outcomes would be: HHTT, HTHT, HTTH, THHT, THTH, TTHH. Here an event, say E, is just an aggregation of all such outcomes. Formally speaking, the event E is defined as the set of all the outcomes corresponding to an interested occurrence. See Fig. 2. Obviously, the event E is a subset of the sample space:

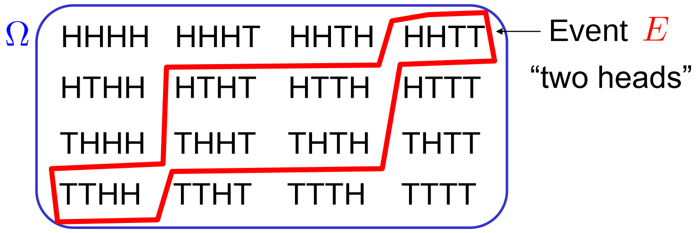


Figure 2: An event E is defined as the set of all the outcomes that respect a condition requested by an interested occurrence, "getting two heads out of four-times coin tossing" in this example.

$E \subseteq \Omega$.

One natural question that arises is then: How should we define the probability of an event E? Naturally we should just *add up* the probabilities of the associated outcomes in E. In other words, for any event $E \subseteq \Omega$, we define the probability of E to be:

$$\mathbb{P}(E) := \sum_{\omega \in E} \mathbb{P}(\omega) \tag{3}$$

where the symbol ":=" means "is defined as". In the prior example in Fig. 1, the probability of getting two heads in four coin tosses can be calculated as:

$$\mathbb{P}(E) = \binom{4}{2} \cdot \frac{1}{16} = \frac{6}{16}. \tag{4}$$

There are $|\Omega| = 16$ possible outcomes for flipping four coins, and each outcome $\omega \in \Omega$ has the same probability $\frac{1}{16}$. As we saw above, there are six ("4-choose-2"$= \binom{4}{2}$) outcomes in E, yielding $\mathbb{P}(E) = \frac{6}{16}$. Here one key observation that we can make is:

Observation : For uniform distribution, the probability calcuation of an event E boils down to *counting* the corresponindg outcomes in E.

Easy examples

In an effort to be familiar with all of the above concepts (sample space, probability model, events), let's exercise ourselves via several examples.

1. Tossing a fair coin once: Here the sample space is obviously $\Omega = \{H, T\}$. Since the coin is *fair*, the probability assignment is straightforward: $\mathbb{P}(\omega) = \frac{1}{2}\ \forall \omega \in \Omega$. An interested event is either "Head" or "Tail". Hence, $\mathbb{P}(H) = \mathbb{P}(T) = \frac{1}{2}$.

2. Tossing still a fair coin yet now *three times*: Here the sample space would be:

$$\Omega = \{HHH, HHT, HTH, HTT, THH, THT, TTH, TTT\}. \tag{5}$$

   Since the coin is still fair, every outcome is equally likely. So the natural probability assignment would be: $\mathbb{P}(\omega) = \frac{1}{|\Omega|} = \frac{1}{8}\ \forall \omega \in \Omega$. Suppose an interested event E is "getting two heads". Then, the probability of E would be:

$$\mathbb{P}(E) = \frac{\binom{3}{2}}{8} = \frac{3}{8} \tag{6}$$

   as there are 3 ($= \binom{3}{2}$) possible outcomes. For a different event like "All three are same", $\mathbb{P}(E) = \frac{1}{8} + \frac{1}{8}$, as there are only two such outcomes: $E = \{HHH, TTT\}$.

3. Tossing now a *biased* coin once: Regardless of the coin characteristic, the sample space is the same of that in the first example: $\Omega = \{H, T\}$. But the bias characteristic of the coin would indeed affect the probability distribution. Suppose that "Head" is twice probable relative to "Tail". Then, the right way to assign probabilities would be:

$$\mathbb{P}(H) = \frac{2}{3}, \quad \mathbb{P}(T) = \frac{1}{3}.$$

   You may want to consider this as the first simplest example in which the outcomes have non-uniform probabilities.

4. Tossing the biased coin yet now *twice*: Again, the sample space is easy to come up with: $\Omega = \{HH, HT, TH, TT\}$. But it is not immediately obvious how to assign probabilities to the outcomes. This is because the bias of the coin only tells us how to assign probabilities to the outcome of *one* flip, not the outcome of *multiple* flips. The only thing that we are 100% sure about is that the probabilities of the outcomes should not be uniform. I understand some of you guys might argue $\mathbb{P}(HH) = \frac{2}{3} \times \frac{2}{3}$. Yes, that's a right way to go. But it is actually based on your intuition and/or your prior knowledge on the concept of *independence* that you learned from high school math. Since we did not learn about the independence yet in this course, let's defer a rigorous discussion to a later lecture when we will dig into details on the independence. Please be patient.

5. Rolling two fair dice: In this case, the sample space reads:

$$\Omega = \{(1,1),(1,2),(1,3),\ldots,(4,6),(5,6),(6,6)\}.$$

Each of the 36 outcomes has equal probability $\frac{1}{36}$. Suppose an interested event E is that the sum of the dice is at least 10. Again, since the probability distribution is uniform, we can compute the probability of E simply via *counting* the interested outcomes:

$$(4,6),(5,5),(5,6),(6,4),(6,5),(6,6).$$

Hence, $\mathbb{P}(E) = \frac{6}{36}$.

A non-trivial example

We have thus far examined only easy examples. So you may be too much confident about the concepts of sample space, probability assignment, and events. But it turns out these concepts are not that simple. There are tons of examples in which it is extremely tricky to come up with a proper sample space and/or to compute the probability of an interested event. You know what? Even expert mathematicians in probability theory often make mistakes in computing the probability of an event due to non-smart handling of the sample space and probability assignment. Frankly speaking, the field of probability is still most difficult to me, although I have been experiencing this field for more than 20 years. Actually, there are lots of smart techniques that help us to easily compute the probability of an event. Here I would like to introduce one of them via a non-trivial example.

The example is a famous one. Consider a situation in which there are 3 students in a classroom and they have their own birthdays. For the sake of simplicity, let us assume that a year has exactly 365 days – forget about "leap year" ("yoon-nien"). First of all, what is the sample space Ω? It would be the set of triplets, each indicating a birthday of each student, say from 1 to 365:

$$\Omega = \{(1,1,1),(1,1,2),(1,1,3),\ldots,(365,365,364),(365,365,365)\}.$$

So, its cardinality is $|\Omega| = 365^3$. Suppose a birthday of each student is equally likely over different 365 days, and it has nothing to do with those of other students. Then, a natural probability assignment would be uniform: $\mathbb{P}(\omega) = \frac{1}{365^3}\ \forall \omega \in \Omega$.

Now consider the following event E: "*At least* two students have the same birthday". Since it is uniform, the only thing that matters is to count such outcomes. How many are such outcomes? There may be: only two students having the same birthday, or all of three having the same. One would expect that there would be many. Actually, if we consider a more general case in which we have n students in a classroom, counting the number of such outcomes would be very much complicated. This is where an interesting trick kicks in. The trick is: whenever you encounter the situation where counting looks complicated (in many cases, the interested event contains a phrase like "at least"), think about its *complement*: $E^c := \{\omega : \omega \in \Omega, \omega \notin E\}$. It turns out this is exactly the case where counting $|E^c|$ is much easier. To see this, consider the complement of E: "No one has the same birthday". How to count such outcomes? Yes, the permutation comes to rescue! There are 365 choices for the first student, 364 choices for the second student (to be distinct with that of the first), and 363 choices for the last. So it is the same as the number of possible ways to select 3 out of 365 choices (here the order matters):

$$|E^c| = {}_{365}P_3 = \frac{365!}{(365-3)!} = 365 \times 364 \times 363 \qquad (7)$$

where $n! := n \times (n-1) \times \cdots \times 2 \times 1$. Using this, we can then easily compute:

$$\mathbb{P}(E) = \frac{|\Omega| - |E^c|}{|\Omega|} = 1 - \frac{{}_{365}P_3}{365^3}. \tag{8}$$

One can also readily extend this to a general case in which there are n students. In this case, the probability of the interested event E would be:

$$\mathbb{P}(E) = \frac{|\Omega| - |E^c|}{|\Omega|} = 1 - \frac{{}_{365}P_{n}}{365^{n}}. \tag{9}$$

Birthday paradox

As mentioned earlier, it is a very famous example – even it has an interesting name. The naming comes from the fact that the exact value of $\mathbb{P}(E)$ goes somewhat against many people's intuition. That's why it is called "Birthday *paradox*". People initially thought that the probability of having two students with the same probability is not that large because there are many candidates (365 days) for a birthday. But the exact computation says: for $n = 23$, $\mathbb{P}(E)$ is over 50%; for $n = 60$, $\mathbb{P}(E)$ is over 99%.

Look ahead

Like I said, there are many non-trivial examples in which computing the probability of an interested event is difficult. Next time, we will investigate one more such example, which is also very famous. The example that I will introduce is the one where coming up with a proper sample space is somewhat highly non-straightforward and we can easily make mistakes if we rush into probability computation while relying upon intuition. From the example, I will then emphasize the importance of the systematic & rigorous approach based on the probability model: first defining a sample space; and then computing the probability of an interested event based on the sample space.

EE210 Probability and Introductory Random Processes — March 9, 2021
KAIST, Spring 2021 — Changho Suh (chsuh@kaist.ac.kr)

Lecture 3: Monty Hall Problem

Recap

Last time, we have studied the most basic concepts in probability: sample space, probability model and events. The sample space Ω is the set of all the outcomes in an experiment. The probability model is a framework comprising the sample space Ω and the probability distribution $\mathbb{P}(\omega)$ subject to the non-negativity and sum-up-to-one constraints:

$$0 \leq \mathbb{P}(\omega) \leq 1 \qquad \text{for all } \omega \in \Omega; \tag{1}$$

$$\sum_{\omega \in \Omega} \mathbb{P}(\omega) = 1. \tag{2}$$

An event E is a subset of Ω that contains certain elements (outcomes) associated with an interested occurrence, and its probability is defined as:

$$\mathbb{P}(E) := \sum_{\omega \in E} \mathbb{P}(\omega). \tag{3}$$

We also exercised ourselves via several examples. At the end of the last lecture, I then claimed that there are many non-trivial counter-intuitive examples in which one can easily make mistakes.

Today's lecture

Today we will investigate one such prominent example. Specifically what we are going to do are four folded. First I will explain the context of the example: how the example comes up in what context. We will then investigate an interested question that was raised in the example. It turns out the interested question can easily be wrongly answered once we rely only upon intuition. Next we will come up with a proper sample space and probability model that lead to the correct answer. Finally I will emphasize the importance of the systematic approach based on the sample space and the probability model.

Figure 1: (Left) Logo of the famous American game show in the past; (Right) Picture of Monty Hall, the original host of the game show. The "Monty Hall Problem" is named after him.

Context

An interesting problem that I would like to discuss in depth in this lecture is the one posed in a famous American television game show, named "Let's Make A Deal" (many people simply called it LMAD at the time). The problem is named after its host: Monty Hall. Hence, it is called the *Monty Hall Problem*. The format of the show is as follows. A person is selected from the audience and the person (called the "trader") makes a deal with the game show host: Monty Hall. The famous Monty Hall Problem was one such problem that arose in the process of a deal.

Monty Hall Problem

Here is the setting of the problem. There are three doors. The prize "car" is behind one door, but this fact is *unknown* to the trader while being revealed to the host. Behind the other two doors are "goats". See Fig. 2.

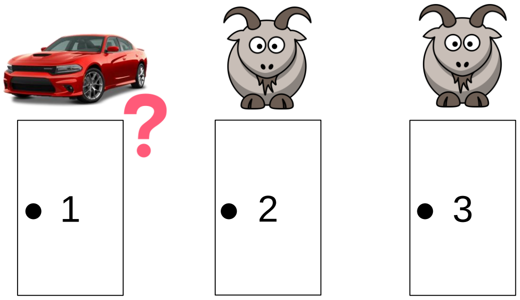


Figure 2: The setting of the Monty Hall Problem. The prize "car" is hidden behind a door, which is unknown to the trader. There are goats (sort of "qquang") behind the other two doors. The host knows about this setting.

Given the setting, a deal begins as follows. First the trader is asked to pick one door in the hope that there is the prize "car" behind the chosen door. The host, who knows which behind each door, opens another door behind which there is a goat, which we simply call "goat-door". One instance is illustrated in Fig. 3. The trader has chosen door 1 (luckily the car-door) and the host opened door 3 (a goat-door). Notice that such goat-door always exists because there are two goat-doors in the setting. Even if the trader picks a goat-door initially (say, door 2 in this example), there is always another goat-door left (door 3 in this case).

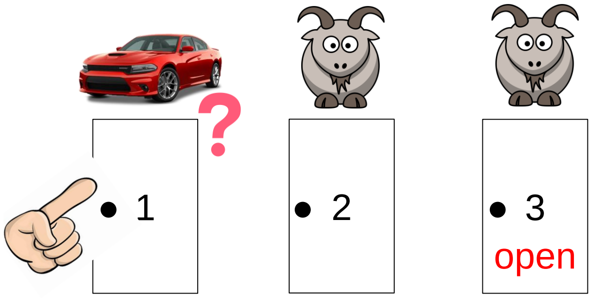


Figure 3: (Step 1) The trader is asked to pick a door for the prize "car"; (Step 2) The host opens another door having a goat; (Step 3) The trader is given a choice between *sticking* with the initial choice vs *switching* to another unopened door.

Next the trader is given a choice between the following two strategies: (i) sticking with his/her initial choice of a door (door 1 in the example); or (ii) switching to another unopened door (door 2 in the example). See Fig. 4. To make a smart choice, of course, the trader has to figure out

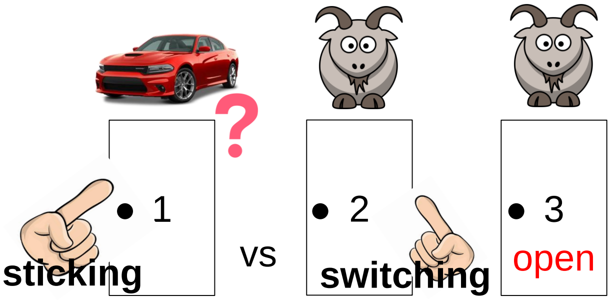


Figure 4: Sticking with the initial choice vs switching to another unopened door.

which strategy is more beneficial. In other words, the following two probabilities are of interest:

$$\mathbb{P}(\text{win w/ sticking}) \qquad \text{vs} \qquad \mathbb{P}(\text{win w/ switching}).$$

Initial reaction of many people

For the first probability, many people at the time could easily come up with an answer:

$$\mathbb{P}(\text{win w/ sticking}) = \frac{1}{3}.$$

There are three doors and the car is behind one door. Hence, given no action (sticking), the winning probability would remain the same as $\frac{1}{3}$. Now what about the second probability? Many people at the time thought that it may be the same as $\frac{1}{3}$ because there is no car-goat locational change (the car and goats did not move at all), so nothing changes:

$$\text{(Initial guess): } \mathbb{P}(\text{win w/ switching}) = \frac{1}{3}. \tag{4}$$

Actually this was also my initial guess when I first encountered this problem from my probability teacher at UC Berkeley, Jean Walrand. Later I realized from wikipedia that even Paul Erdős, one of the most prolific mathematicians in history, believed so.

Switching is more beneficial!

As I hinted before, however, you can easily imagine that it is not the case. I told you this is a non-trivial & counter-intuitive example. So the answer must be: the second probability is not the same as $\frac{1}{3}$. It turns out it is indeed the case. Now then, you may wonder what the second probability is. To figure this out, you may be forced to think about something changed between two timings: (i) the initial timing; (ii) the later timing when an option is given to the trader (between sticking vs switching). The key distinction is that in the later timing, a goat-door is opened by the host. In this case, the opened door is eliminated for a choice. Hence, this may naturally motivate you to guess about the second probability as:

$$\text{(Second guess): } \mathbb{P}(\text{win w/ switching}) = \frac{1}{2}, \tag{5}$$

as there are only two doors left and one is a goat-door while the other is the car-door for sure. Actually this was exactly my second guess when Jean Walrand told me that my initial guess was wrong. It turns out this guess is still wrong. You may wonder what happened. That was also my initial reaction – I felt very unconfident about my probability skills at the time. It turns out the correct answer is that the winning probability with switching is twice larger than that with sticking:

$$\text{(Correct answer): } \mathbb{P}(\text{win w/ switching}) = \frac{2}{3}. \tag{6}$$

From next sections onwards, I will provide a rigorous proof of (6).

Sample space

For a rigorous proof, we need to rely upon *sample space* instead of intuition which can potentially be very shaky. Now how to construct a sample space? To this end, we first need to think about where uncertainty comes up. There are three sources of uncertainty in the problem setting: (i) car's location; (ii) trader's initial choice; (iii) host's choice. So one can think of the following triplet:

$$(\text{car's location}, \text{trader's choice}, \text{host's choice}). \tag{7}$$

A next question is then: What are the possible triplets? To easily come up with these, let's consider two cases:

$$\begin{aligned}&\text{(Case I): car's location} = \text{trader's choice;}\\&\text{(Case II): car's location} \neq \text{trader's choice.}\end{aligned}$$

In Case I (e.g., car's location and trader's choice is door 1 as in Fig. 5(Left)), there are two goat-doors left. So host's choice would be either door 2 or door 3. The following six triplets are of this category:

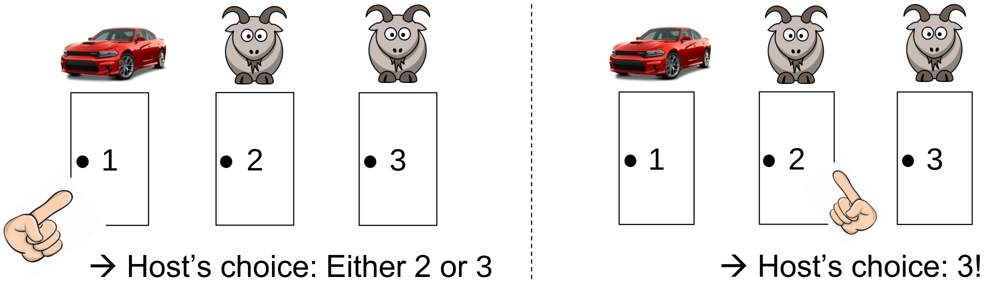


Figure 5: (Left) Case I: car's location = trader's choice; (Right) Case II: car's location ≠ trader's choice.

$$(1,1,2), (1,1,3), (2,2,3), (2,2,1), (3,3,1), (3,3,2).$$

In Case II (e.g., car's location is door 1 while trader's choice is door 2 as in Fig. 5(Right)), there is only one goat-door left. Hence, host's choice must be door 3! There are six such triplets. Why? Think about ${}_3P_2 = 3 \times 2 = 6$. Hence, the followings are of this category:

$$(1,2,3), (2,1,3), (1,3,2), (3,1,2), (2,3,1), (3,2,1).$$

Aggregating all of the above possible triplets, we construct a sample space as:

$$\begin{aligned}\Omega = \{&{\color{blue}(1,1,2), (1,1,3), (2,2,3), (2,2,1), (3,3,1), (3,3,2)}\\&{\color{red}(1,2,3), (2,1,3), (1,3,2), (3,1,2), (2,3,1), (3,2,1)}\}.\end{aligned} \tag{8}$$

Probability distribution

The next thing to do is to come up with the probability distribution: $\mathbb{P}(\omega)$ for all $\omega \in \Omega$. To this end, consider all the possible configurations that depend on car's location and trader's choice. Since there are three possible choices for each, we have 9 $(= 3 \times 3)$ cases in total. These choices are random. Hence, one can assume that such 9 cases are equi-probable. This motivates us to construct the probability distribution as:

$$\mathbb{P}((1,2,3)) = \mathbb{P}((2,1,3)) = \mathbb{P}((1,3,2)) = \mathbb{P}((3,1,2)) = \mathbb{P}((2,3,1)) = \mathbb{P}((3,2,1)) = \frac{1}{9}; \tag{9}$$

$$\mathbb{P}((1,1,2)) + \mathbb{P}((1,1,3)) = \mathbb{P}((2,2,3)) + \mathbb{P}((2,2,1)) = \mathbb{P}((3,3,1)) + \mathbb{P}((3,3,2)) = \frac{1}{9}. \tag{10}$$

Here we add two probabilities when car's location is the same as trader's choice, e.g., $\mathbb{P}((1,1,2)) + \mathbb{P}((1,1,3)) = \frac{1}{9}$. The reason is that for such case, there are two sub-cases. This is also illustrated in Fig. 6.

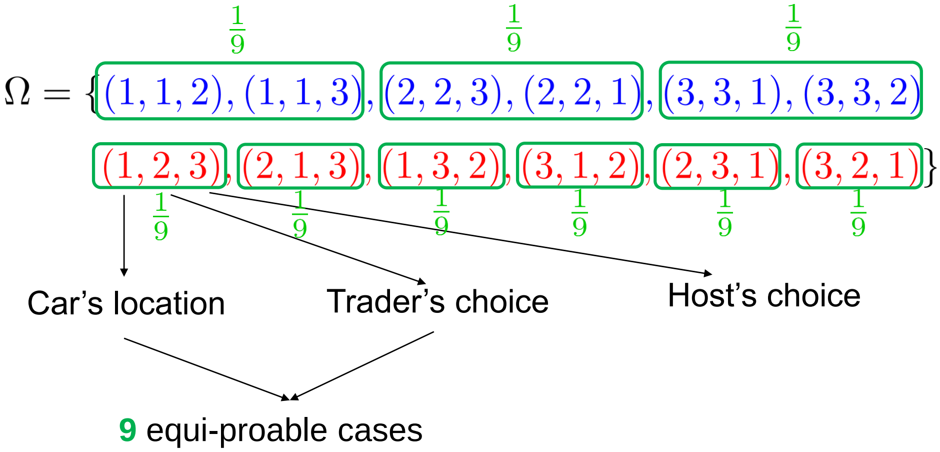


Figure 6: Probability distribution: Depending on random choices of car's location and trader's choice, there are 9 $(= 3 \times 3)$ equi-probable cases. Hence, the probability $\frac{1}{9}$ is assigned to each of the 9 cases. For instance, $\mathbb{P}((1,2,3)) = \frac{1}{9}$. But there are two sub-cases when car's location is the same as trader's choice. So in this case, we map: $\mathbb{P}((1,1,2)) + \mathbb{P}((1,1,3)) = \frac{1}{9}$.

Now let us assume that if car's location is the same as trader's choice, then host's choice is random between two goat-doors left. We can then assign equal probabilities for two sub-cases:

$$\mathbb{P}((1,1,2)) = \mathbb{P}((1,1,3)) = \mathbb{P}((2,2,3)) = \mathbb{P}((2,2,1)) = \mathbb{P}((3,3,1)) = \mathbb{P}((3,3,2)) = \frac{1}{18}. \tag{11}$$

Computation of $\mathbb{P}$(win w/ switching)

We are now ready to prove (6). The event of winning with switching corresponds to all the triplets where car's location differs from trader's choice (marked in red) in the sample space Ω. This together with (9) yields:

$$\begin{aligned}\mathbb{P}(\text{win w/ switching}) &= \mathbb{P}((1,2,3)) + \mathbb{P}((2,1,3)) + \mathbb{P}((1,3,2)) \\ &\quad + \mathbb{P}((3,1,2)) + \mathbb{P}((2,3,1)) + \mathbb{P}((3,2,1)) = \frac{6}{9} = \frac{2}{3}.\end{aligned}$$

We can also double-check the sticking-winning probability that we guessed as $\frac{1}{3}$ before. The winning event under this sticking strategy corresponds to all the triplets where car's location is the same as trader's initial choice (marked in blue). Hence, we get:

$$\begin{aligned}\mathbb{P}(\text{win w/ sticking}) &= \mathbb{P}((1,1,2)) + \mathbb{P}((1,1,3)) + \mathbb{P}((2,2,3)) \\ &\quad + \mathbb{P}((2,2,1)) + \mathbb{P}((3,3,1)) + \mathbb{P}((3,3,2)) = \frac{6}{18} = \frac{1}{3}.\end{aligned}$$

Lesson

Before concluding this lecture, I would like to emphasize one lesson that I learned from the Monty Hall Problem. That is,

Don't rely solely upon intuition. Go back to the basics if you feel shaky.

The reason that I was wrong for the above two guesses ((4) and (5)) was that I relied only upon my intuition and never tried to compute the probability based on the probability model. So I realized that if I am not 100% sure about my intuition, then I should dig into the probability model and try to get a reliable answer out of it.

In fact, there is one more thing that you can do even after a rigorous proof. That is to confirm the proof with a *computer simulation*. You know what? Often times, some very smart and ego-centric experts do not believe other people's proof although the proof looks rigorous. The only way to convince such stubborn guys is to show them a simulation result confirming the proof. Actually Paul Erdős, the math hero in history, did not trust in the proof of (6) until he was shown a confirming computer simulation. This is one of the reasons that you should be very good at programming. In an effort to help you out, I made one coding exercise problem in PS1 where you can confirm (6) with a Python code simulation.

Look ahead

So far we have studied the basic concepts of sample space, probability model and events, with the help of several examples. Next time, we will study another set of basis concepts in probability: conditional probability, total probability law and Bayes' law.

EE210 Probability and Introductory Random Processes March 11, 2021
KAIST, Spring 2021 Changho Suh (chsuh@kaist.ac.kr)

Lecture 4: Conditional probability & total probability law

Recap

Last time, we investigated one prominent yet counter-intuitive problem in which one (even experts) can easily make mistakes: the Monty Hall Problem. One lesson that I wished to deliver from the problem was that: "Don't rely solely upon intuition. Instead go back to the basics if you feel shaky." In the context of probability, what the lesson means is that: "First construct a sample space, then come up with the corresponding probability distribution; lastly compute the probability of an interested event based on the probability model."

Today's lecture

Today we will move onto other important basic concepts in probability: conditional probability, total probability law, and Bayes' law. Specifically what we are going to cover are five folded. First we will introduce the definition of conditional probability. I will then explain the rationale behind the definition. Next we will exercise ourselves on the learned concept with one very popular problem, named the *disease testing.* In fact, there is another relevant theorem that arises in the process of solving the problem: *total probability law.* So in the fourth part, we will study the law. There is another important and useful theorem used in the context of the disease testing problem: *Bayes' law.* In the last part, we will discuss it.

Definition of conditional probability

Conditional probability is defined with respect to (w.r.t.) multiple events. For simplicity, consider a situation where we are interested in two events, say A and B. The probability of A conditioned on B is denoted by $\mathbb{P}(A|B)$ and defined as:

$$\mathbb{P}(A|B) := \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)}. \tag{1}$$

Here one thing that we should be very careful about is that the notation $\mathbb{P}(\cdot)$ used in the right hand side is different from $\mathbb{P}(\cdot|\cdot)$ used in the left hand side, although we employ the same notation $\mathbb{P}$. Of course, these two can be differentiated since the number of arguments is different: $\mathbb{P}(\cdot)$ takes one argument while $\mathbb{P}(\cdot|\cdot)$ takes two. To differentiate these more clearly, one may use the following notations:

$$\mathbb{P}_B(A) = \mathbb{P}(A|B), \quad \mathbb{P}_\Omega(A) = \mathbb{P}(A). \tag{2}$$

Here we put the sample space in the subscript of $\mathbb{P}$. In the case of $\mathbb{P}(\cdot)$, the sample space is Ω, while for $\mathbb{P}(\cdot|\cdot)$, the sample space becomes restricted to the particular event B, as B already happened (that's what it means by being conditioned on). But the convention is not to use such subscript notation. So we will take the convention to employ the same notation $\mathbb{P}$:

$$\mathbb{P}(A|B), \quad \mathbb{P}(A). \tag{3}$$

Again, these two can be differentiated from the different number of arguments taken.

Rationale behind the definition $\mathbb{P}(A|B) := \frac{\mathbb{P}(A\cap B)}{\mathbb{P}(B)}$

Now you may wonder why we define conditional probability like (1). There is always a good reason behind any definition. The reason can readily be explained from a Venn diagram as illustrated in Fig. 1. Here one key observation is that conditioned on B, the event B becomes the *new sample space*. Since the event already happened, any outcome outside the event never occurred.

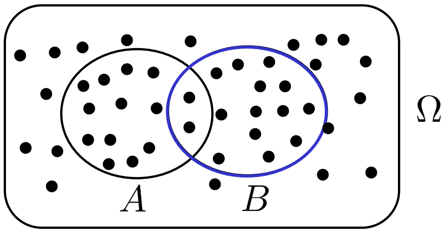


Figure 1: Conditioned on the event B: B becomes the *new sample space*, as the event already occurred. We do not need to worry about other outcomes lying in B^c.

Given the new sample space, the next thing to do is to come up with the corresponding probability distribution. Let $\mathbb{P}(\omega|B)$ be the probability distribution w.r.t. the new sample space. Remember that there is the sum-up-to-one constraint on the probability assignment:

$$\sum_{\omega \in B} \mathbb{P}(\omega|B) = 1. \tag{4}$$

Now how to define $\mathbb{P}(\omega|B)$? The following observation gives a hint. The probability of B reads:

$$\sum_{\omega \in B} \mathbb{P}(\omega) = \mathbb{P}(B). \tag{5}$$

Dividing both sides by $\mathbb{P}(B)$, we get:

$$\sum_{\omega \in B} \frac{\mathbb{P}(\omega)}{\mathbb{P}(B)} = 1. \tag{6}$$

Looking at (4) and (6), one natural definition for $\mathbb{P}(\omega|B)$ is:

$$\mathbb{P}(\omega|B) := \frac{\mathbb{P}(\omega)}{\mathbb{P}(B)}. \tag{7}$$

Only in light of (4) and (6), there could be other ways to define $\mathbb{P}(\omega|B)$, as the equality for summation does not necessarily imply the equality for every individual participating in the summation. But normalizing $\mathbb{P}(\omega)$ by $\mathbb{P}(B)$ like (7) is also the way to preserve the probability behaviour w.r.t. ω. Hence, people take it as the definition.

Now in the new sample space B, the probability of A is calculated as:

$$\mathbb{P}(A|B) = \sum_{\omega \in A \cap B} \mathbb{P}(\omega|B). \tag{8}$$

This is because the event A conditioned on B reads $A \cap B$; also see Fig. 2. Applying the

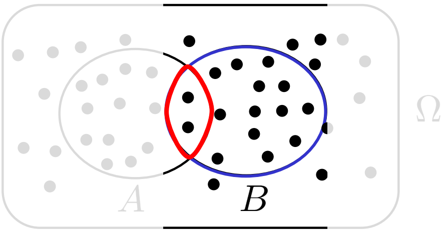


Figure 2: Conditioned on B, the occurrence of A can be interpreted as the intersection between A and B.

definition (7) to (8), we get:

$$\begin{aligned}
\mathbb{P}(A|B) &= \sum_{\omega \in A \cap B} \mathbb{P}(\omega|B) \\
&:= \sum_{\omega \in A \cap B} \frac{\mathbb{P}(\omega)}{\mathbb{P}(B)} \\
&= \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)}
\end{aligned} \tag{9}$$

where the last equality comes from the definition of $\mathbb{P}(A \cap B)$. We see that this exactly coincides with the definition (1) of conditional probability.

A very popular problem: the disease testing problem

Let us explore one very prominent problem where the concept of conditional probability plays a crucial role. The problem is so called the *disease testing problem.* To get a concrete feel, let us consider the *cancer* testing problem.

Suppose that the cancer test of a person reads "positive". Then, the person would obviously be very interested in the correctness of the test. The person would want to believe that the test result is wrong. In an effort to verify the correctness of the test, the person often wants to figure out:

Probability that the person has indeed cancer given the test result is positive.

This is where conditional probability kicks in. Why? Letting A be the event that the person has indeed cancer and B be the event that the test result is positive, we can interpret the interested probability as $\mathbb{P}(A|B)$.

Now how to compute $\mathbb{P}(A|B)$? Recalling the definition:

$$\mathbb{P}(A|B) := \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)}, \tag{10}$$

we immediately see that we need to compute two quantities: $\mathbb{P}(A \cap B)$ and $\mathbb{P}(B)$.

Computation of $\mathbb{P}(A \cap B)$

Let us first consider the computation of the numerator. Here comes a challenge though. The challenge is that the ground truth of such probability $\mathbb{P}(A \cap B)$ is unknown. But there is a good news. The good news is that we can somehow infer the quantity from *statistical data* based on *clinical trials.* In order to understand what it means, see Fig. 3. In clinical trials, one can often

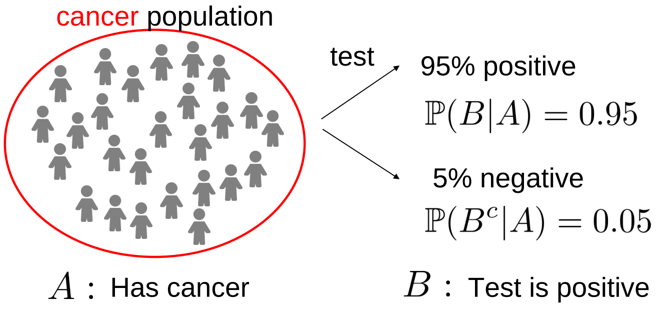


Figure 3: In clinical trials, one can obtain test results for a certain population, say cancer population, as shown in this example. Suppose that around 95% of the tests are positive. Then, one can use this positive rate as the ground-truth of the conditional probability $\mathbb{P}(B|A) = 0.95$. The negative rate 0.05 can also be used as its flipped counterpart $\mathbb{P}(B^c|A) = 0.05$.

obtain test results given a certain population. For instance, suppose that tested people are all from cancer population as in Fig. 3. We test many times for many of such people and gather test results. Suppose that around 95% of the tested people read "positive". Then, what we can say from this is that a good estimate of $\mathbb{P}(B|A)$ might be the fraction 0.95. The estimate would be more and more accurate as the number of tested people grows. In reality, such tests are often done for the purpose of measuring the correctness of the test: the larger the faction is, the more accurate the test is. In the above case, 95% accuracy (often called the true positive rate, TPR for short) and 5% misdetection rate (or called the false negative rate, FNR for short).

Now are we done for the computation of $\mathbb{P}(A \cap B)$ with a good estimate of $\mathbb{P}(B|A)$? Not yet. To see this, using the definition of conditional probability, let us rewrite $\mathbb{P}(A \cap B)$ as:

$$\mathbb{P}(A \cap B) = \mathbb{P}(A)\mathbb{P}(B|A).$$

Here the problem is that we also need to figure out $\mathbb{P}(A)$. Again, statistical data comes to rescue. Statistical data for the cancer-vs-normal populations are often available in the real world. So we can use them to come up with a good estimation of $\mathbb{P}(A)$. For instance, suppose statistics says that 10% of the entire population have cancer. Then, one can somehow assume $\mathbb{P}(A) = 0.1$. Using this together with the above estimate of $\mathbb{P}(B|A)$, we can then compute:

$$\begin{aligned}\mathbb{P}(A \cap B) &= \mathbb{P}(A)\mathbb{P}(B|A) \\ &= 0.1 \times 0.95.\end{aligned} \tag{11}$$

Computation of $\mathbb{P}(B)$

Next consider the computation of the denominator $\mathbb{P}(B)$ in (10). The computation of $\mathbb{P}(B)$ is a bit tricky. First of all, similar to $\mathbb{P}(B|A)$, we can also obtain a good estimate of $\mathbb{P}(B|A^c)$ now by relying upon clinical data obtained from testing on the normal population A^c. Suppose the fraction of test results being positive on the normal population reads 0.2 (called the false alarm rate or false positive rate, FPR for short). Then, the number 0.2 can serve as an estimate of $\mathbb{P}(B|A^c)$. Hence, in reality, $\mathbb{P}(B|A)$ and $\mathbb{P}(B|A^c)$ are estimable from testing. However, the interested quantity $\mathbb{P}(B)$ is not directly available. But there is an *indirect* way to compute $\mathbb{P}(B)$. The way is based on the *total probability law*!

Total probability law

The total probability law is extremely simple yet powerful. Let us first explain what it says. First manipulate the event B as:

$$B = (A \cap B) \cup (A^c \cap B).$$

This is immediate from the Venn diagram in Fig. 4. Here a key observation is that the two individual events $A \cap B$ and $A^c \cap B$ are *disjoint*. This is obvious because A and A^c are disjoint. Hence,

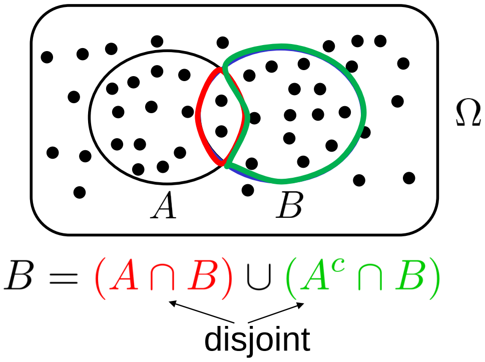


Figure 4: Total probability law: an event B can be represented as the union of $A \cap B$ and $A^c \cap B$. Since these subsets are distinct, $\mathbb{P}(B)$ is simply the sum of the individual probabilities: $\mathbb{P}(B) = \mathbb{P}(A \cap B) + \mathbb{P}(A^c \cap B)$.

$$\mathbb{P}(B) = \mathbb{P}(A \cap B) + \mathbb{P}(A^c \cap B). \tag{12}$$

Why? Think about the definition of the probability of an event. The equality (12) is exactly what the *total probability law* says.

Using the total probability law (12) together with the above estimates of $\mathbb{P}(B|A)$, $\mathbb{P}(B|A^c)$ and $\mathbb{P}(A)$, we can now compute the interested probability:

$$\begin{aligned} \mathbb{P}(B) &= \mathbb{P}(A \cap B) + \mathbb{P}(A^c \cap B) \\ &= \mathbb{P}(A)\mathbb{P}(B|A) + \mathbb{P}(A^c)\mathbb{P}(B|A^c) \\ &= 0.1 \times 0.95 + 0.9 \times 0.2 \end{aligned} \tag{13}$$

where the second equality is due to the definition of conditional probability and the last comes from the estimates.

Computation of $\mathbb{P}(A|B)$

We are now ready to compute the conditional probability $\mathbb{P}(A|B)$. Applying (11) and (13) into (10), we get:

$$\begin{aligned} \mathbb{P}(A|B) := \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)} &= \frac{\mathbb{P}(A)\mathbb{P}(B|A)}{\mathbb{P}(A)\mathbb{P}(B|A) + \mathbb{P}(A^c)\mathbb{P}(B|A^c)} \\ &= \frac{0.1 \times 0.95}{0.1 \times 0.95 + 0.9 \times 0.2} \approx 0.3455. \end{aligned} \tag{14}$$

Discussion on $\mathbb{P}(A|B)$

Notice that the probability (14) reads around 35%. This number can somewhat relieve the person. Why? Because the probability $\mathbb{P}(A|B)$ that the person has indeed the cancer is less than 50% although the test result reads "positive". Actually, this is often the case in reality: even under the positive test result, it is more likely that the tested people are normal. This is one of the main reasons why many people who got positive on cancer testing are trying to get another test from a different hospital.

Then, you may wonder why we have a small value of $\mathbb{P}(A|B)$ in reality. This is because the cancer population A itself is very small, 10% in this example. The smaller the cancer population is, the smaller the interested probability $\mathbb{P}(A|B)$ is. This is somehow well reflected in the definition of conditional probability:

$$\mathbb{P}(A|B) := \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)} = \mathbb{P}(A) \cdot \frac{\mathbb{P}(B|A)}{\mathbb{P}(B)} \tag{15}$$

where the second equality comes from $\mathbb{P}(B|A) := \frac{\mathbb{P}(A\cap B)}{\mathbb{P}(A)}$. Here we see $\mathbb{P}(A)$ is multiplied in front. The precise look says that $\mathbb{P}(A)$ also affects $\mathbb{P}(B)$ placed in the denominator, as $\mathbb{P}(B) = \mathbb{P}(A)\mathbb{P}(B|A) + \mathbb{P}(A^c)\mathbb{P}(B|A^c)$. But the decrease of $\mathbb{P}(A)$ does not directly decrease $\mathbb{P}(B)$ due to the other term $\mathbb{P}(A^c)\mathbb{P}(B|A^c)$. It turns out the term $\mathbb{P}(A)$ placed in front plays a more dominant role in changing $\mathbb{P}(A|B)$. Hence, the smaller $\mathbb{P}(A)$, the smaller $\mathbb{P}(A|B)$. From this example, we can also see that the concept of conditional probability provides such interesting interpretation.

Bayes' law

Lastly I would like to put an emphasis on a very simple yet powerful law that arose in the process of computing $\mathbb{P}(A|B)$: the *Bayes' law*. What the law says is extremely simple. It is just a consequence of applying the definition of conditional probability *twice*:

$$\mathbb{P}(A|B) := \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)} = \frac{\mathbb{P}(A)\mathbb{P}(B|A)}{\mathbb{P}(B)}. \tag{16}$$

where the second equality comes from $\mathbb{P}(B|A) := \frac{\mathbb{P}(A\cap B)}{\mathbb{P}(A)}$. Since this is too immediate, you may wonder why we care about this seemingly-trivial law. Of course, there is a reason. The reason is that the application of the Bayes' law is pretty wide. It turns out there are many interesting scenarios where one wants to compute $\mathbb{P}(A|B)$ but one is given $\mathbb{P}(B|A)$ instead, i.e., only its flipped version is available. You will check such examples in homework.

Look ahead

The main focus of this lecture is the concept of *conditional probability*. There is a very natural follow-up concept: *independence*. Next time, we will study the concept of independence in depth.

EE210 Probability and Introductory Random Processes March 16, 2021
KAIST, Spring 2021 Changho Suh (chsuh@kaist.ac.kr)

Lecture 5: Independence

Recap

Last time, we have studied one important concept: *conditional probability* defined as below:

$$\mathbb{P}(A|B) := \frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)}. \tag{1}$$

I also explained the rationale behind this definition, using the new sample space B (the sample space after being conditioned on B) together with the corresponding probability distribution denoted by $\mathbb{P}(\omega|B)$ where $\omega \in B$. We then studied one very relevant and popular problem: *disease testing problem* tailored for one particular disease, cancer. While we were investigating how conditional probability plays a role in solving the cancer testing problem, I also emphasized two important laws that arose in the process of computing an interested probability in the problem context. One is the *total probability law* which allows us to easily compute the denominator in (**??**):

$$\mathbb{P}(B) = \mathbb{P}(A \cap B) + \mathbb{P}(A^c \cap B). \tag{2}$$

The proof of this is obvious. This is because of the following two: (i) two events $A \cap B$ and $A^c \cap B$ are *disjoint*; (ii) $\mathbb{P}(B)$ is computed by aggregating over all $\omega \in B = (A \cap B) \cup (A^c \cap B)$. The second is the *Bayes' law*:

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(A)\mathbb{P}(B|A)}{\mathbb{P}(B)}. \tag{3}$$

The proof of this is also immediate. It is just a consequence of applying the definition of conditional probability twice. I told you that Bayes' law is beneficial particularly when one wants to compute $\mathbb{P}(A|B)$ but one is given $\mathbb{P}(B|A)$ instead.

Today's lecture

Today we will go forward, exploring another follow-up and very relevant concept: *independence*. Specifically what we are going to do are four folded. First we will introduce the definition of independence for a simple setting in which only two events are taken into consideration. I will then explain the rationale behind the definition under the simple context. Next we will generalize the definition to more-than-two-events cases. As we did during the past lectures, in the last part, we will exercise ourselves on the learned concept with a couple of examples.

Definition of independence for two events

First consider a simple setting where we are interested in two events, say A and B. We say that events A and B are *independent* if the probability of the intersected event is the product of individual probabilities:

$$\mathbb{P}(A \cap B) = \mathbb{P}(A)\mathbb{P}(B). \tag{4}$$

You may wonder why we define the independence as above. Of course, there must be a reason.

Rationale behind the definition (??)

The reason is obviously related to a natural definition that we can think of for independence. In a natural sense, we should say that A and B are independent if the probability of A *has nothing to do with* whether or not the event B occurs, i.e., $\mathbb{P}(A)$ remains the same whether B is conditioned on:

$$\mathbb{P}(A|B) = \mathbb{P}(A). \tag{5}$$

Using the definition of conditional probability, we can then rewrite the natural condition (**??**) as:

$$\frac{\mathbb{P}(A \cap B)}{\mathbb{P}(B)} = \mathbb{P}(A). \tag{6}$$

We see this is exactly the same as the original definition (**??**): $\mathbb{P}(A \cap B) = \mathbb{P}(A)\mathbb{P}(B)$.

Now you may wonder why we do not employ the above more intuitive condition (**??**) instead of (**??**). The reason is that in such a way, we need another condition:

$$\mathbb{P}(B) > 0,$$

as otherwise $\mathbb{P}(A|B)$ is not definable. Or we need $\mathbb{P}(A) > 0$ if we take the flipped-around condition $\mathbb{P}(B|A) = \mathbb{P}(B)$. On the other hand, in the original definition (**??**), we do not need such conditions, which is obviously better.

Definition for the three events case

Next let's think about the three-events case where we have say A, B and C. In this case, we say that A, B and C are *mutually independent* if any paired intersected event is the product of the individual probabilities and similarly for the tripled intersected event $A \cap B \cap C$:

$$\begin{aligned} \mathbb{P}(A \cap B) &= \mathbb{P}(A)\mathbb{P}(B); \\ \mathbb{P}(B \cap C) &= \mathbb{P}(B)\mathbb{P}(C); \\ \mathbb{P}(C \cap A) &= \mathbb{P}(C)\mathbb{P}(A); \\ \mathbb{P}(A \cap B \cap C) &= \mathbb{P}(A)\mathbb{P}(B)\mathbb{P}(C). \end{aligned} \tag{7}$$

The rationale behind this definition is the same as before. This comes from the naturally-looking conditions expressed in terms of conditional probability and its unconditional counterpart. See below. Here the naturally-looking conditions try to indicate that the occurrence of any number of the events carries no information on the remaining events. In Fig. **??**, we list *only part* of such conditions. Notice that $\mathbb{P}(A|B \cap C) = \mathbb{P}(A)$ together with $\mathbb{P}(B|C) = \mathbb{P}(B)$ yields $\mathbb{P}(A \cap B \cap C) = \mathbb{P}(A)\mathbb{P}(B)\mathbb{P}(C)$. Of course, there are a bunch of other ways. For instance,

$$\begin{tikzcd}
\mathbb{P}(A \cap B) = \mathbb{P}(A)\mathbb{P}(B) & \mathbb{P}(A|B) = \mathbb{P}(A) \arrow[l] \\
\mathbb{P}(B \cap C) = \mathbb{P}(B)\mathbb{P}(C) & \mathbb{P}(B|C) = \mathbb{P}(B) \arrow[l] \\
\mathbb{P}(C \cap A) = \mathbb{P}(C)\mathbb{P}(A) & \mathbb{P}(C|A) = \mathbb{P}(C) \arrow[l] \\
\mathbb{P}(A \cap B \cap C) = \mathbb{P}(A)\mathbb{P}(B)\mathbb{P}(C) & \mathbb{P}(A|B \cap C) = \mathbb{P}(A) \arrow[l]
\end{tikzcd}$$

Figure 1: The definition of independence is motivated from the natural conditions (marked in red) expressed in terms of conditional probability and its unconditional counterpart.

$\mathbb{P}(A \cap B|C) = \mathbb{P}(A \cap B)$ together with $\mathbb{P}(A|B) = \mathbb{P}(B)$ yields $\mathbb{P}(A \cap B \cap C) = \mathbb{P}(A)\mathbb{P}(B)\mathbb{P}(C)$. From this, we see another good thing about the original definition (**??**): its expression is more succinct than the one based on the naturally-looking conditions.

Another important thing to mention here is that we call such independence, *mutual independence*, instead of just independence. This is because there is another different definition of independence, named the *pairwise independence*. We say that A, B and C are *pairwise* independent if only the paired intersected events are the product of their individuals:

$$\begin{aligned} \mathbb{P}(A \cap B) &= \mathbb{P}(A)\mathbb{P}(B); \\ \mathbb{P}(B \cap C) &= \mathbb{P}(B)\mathbb{P}(C); \\ \mathbb{P}(C \cap A) &= \mathbb{P}(C)\mathbb{P}(A). \end{aligned} \tag{8}$$

It turns out there are many cases in which events are pairwise independent, but not mutually independent. See one such example in the next section.

On a different note, some curious students may wonder whether the following condition only

$$\mathbb{P}(A \cap B \cap C) = \mathbb{P}(A)\mathbb{P}(B)\mathbb{P}(C) \text{ (that we call the \textit{three-way independence})} \tag{9}$$

implies the pairwise independence of any events. It turns out this does not hold either. There are many contrived counter-examples. See one such example in the next next section.

At this point, you may be very confused about the concept of independence, from three events onwards. Yes, it is indeed confusing. Here is my recommendation which may serve you to be less confused. Just remember the *original definition* (**??**) and check whether all the conditions therein are satisfied – don't be bothered by others. This is another reason that you should *memorize the definition*. Indeed, the mathematics is a field of *memorization*.

Example: Pairwise independent but not mutually independent

Consider an experiment of flipping a fair coin twice. Let A and B be the events that the 1st and 2nd flips show "Head" and "Tail", respectively. Let C be the event that the 1st and 2nd flips are different. Notice that the occurrence of the event C is determined by the events A and B. Hence, you intuition says these are not mutually independent. Yes, you are right. Let's also verify that via the *definition* (**??**). First compute:

$$\begin{aligned} \mathbb{P}(A \cap B \cap C) &= \mathbb{P}(A \cap B)\mathbb{P}(C|A \cap B) \\ &= \mathbb{P}(A \cap B) \\ &= \frac{1}{4} \end{aligned} \tag{10}$$

where the 1st equality is due the definition of conditional probability; the 2nd equality comes from the fact that given A and B, the event C must occur, i.e., $\mathbb{P}(C|A \cap B) = 1$; and the last is because HT is one of the four events $\{HH, HT, TH, TT\}$ equally likely. On the other hand,

$$\mathbb{P}(A)\mathbb{P}(B)\mathbb{P}(C) = \frac{1}{2} \times \frac{1}{2} \times \frac{1}{2} = \frac{1}{8} \neq \frac{1}{4}. \tag{11}$$

Here, the event C is associated with $\{HT, TH\}$ out of four; hence, $\mathbb{P}(C) = \frac{2}{4} = \frac{1}{2}$. From this, we see that (A, B, C) are not mutually independent.

But these are *pairwise* independent. To see this, we first check:

$$\mathbb{P}(A \cap B) = \frac{1}{4} = \frac{1}{2} \times \frac{1}{2} = \mathbb{P}(A)\mathbb{P}(B). \tag{12}$$

What about A and C? We first compute:

$$\begin{aligned}\mathbb{P}(A \cap C) &= \mathbb{P}(A)\mathbb{P}(C|A) \\ &= \frac{1}{2}\mathbb{P}(B|A) \\ &= \frac{1}{2}\mathbb{P}(B) \\ &= \frac{1}{2} \times \frac{1}{2}\end{aligned} \tag{13}$$

where the 2nd equality comes from the fact that conditioned on A (1st flip is "Head"), the event C (two are different) is equivalent to B (2nd flip is "Tail"); and the 3rd is because of the independence of A and B. As we computed earlier, $\mathbb{P}(C) = \frac{1}{2}$. This together with $\mathbb{P}(A) = \frac{1}{2}$ and (**??**) proves the independence of A and C. By symmetry, one can also prove the independence of B and C. Hence, (A, B, C) are *pairwise* independent.

Example: Three-way independent but not pairwise independent

Let's consider a contrived and highly non-intuitive example where the three-way independence does not necessarily imply the pairwise independence. Consider an experiment of rolling two fair dice, and the following three events:

$$\begin{aligned}A &= \{\text{1st shows } 1, 2 \text{ or } 3\}; \\ B &= \{\text{1st shows } 3, 4 \text{ or } 5\}; \\ C &= \{\text{the sum of the two dice is } 9\}.\end{aligned} \tag{14}$$

The event $A \cap B \cap C$ is associated with $(3, 6)$ out of 36 events equally likely, so $\mathbb{P}(A \cap B \cap C) = \frac{1}{36}$. We also compute: $\mathbb{P}(A) = \frac{3}{6} = \frac{1}{2}$, $\mathbb{P}(B) = \frac{3}{6} = \frac{1}{2}$, and $\mathbb{P}(C) = \frac{4}{36} = \frac{1}{9}$ $(\{(3, 6), (4, 5), (5, 4), (6, 3)\})$. Hence,

$$\mathbb{P}(A \cap B \cap C) = \frac{1}{36} = \frac{1}{2} \times \frac{1}{2} \times \frac{1}{9} = \mathbb{P}(A)\mathbb{P}(B)\mathbb{P}(C). \tag{15}$$

On the other hand, these are not pairwise independent. For instance,

$$\mathbb{P}(A \cap B) = \frac{1}{6} \neq \frac{1}{2} \times \frac{1}{2} = \mathbb{P}(A)\mathbb{P}(B).$$

Mutual independence for n events

Now consider a really general case in which there are an arbitrary number of events, say A_1, all the way up to, A_n. We say that the events $(A_1, \ldots, A_n)$ are *mutually independent* if for every subset $I \subseteq \{1, 2, \ldots, n\}$, the probability of the intersection of the events associated with I is the product of the corresponding individuals:

$$\mathbb{P}\left(\bigcap_{i \in I} A_i\right) = \prod_{i \in I} \mathbb{P}(A_i) \tag{16}$$

where $\bigcap_{i \in I} A_i := A_{i_1} \cap \cdots \cap A_{i_{|I|}}$, $\prod_{i \in I} \mathbb{P}(A_i) := \mathbb{P}(A_{i_1}) \cdots \mathbb{P}(A_{i_{|I|}})$ and $(i_1, \ldots, i_{|I|})$ are the elements of I. The rationale behind this definition is exactly the same as the one in the three-events case. Again, my recommendation is to just remember this definition and apply.

We are done with the definition of independence. As mentioned in the beginning, let's now exercise ourselves on the concept of independence via a couple of examples. We will investigate four examples in total.

Example #1: Tossing the $\frac{2}{3}$-biased coin twice

The first example is the one that we already explored in Lecture 2 only with intuition yet not with a rigorous step. That is the experiment of tossing a biased coin (with $\frac{2}{3}$ probability of showing "Head") twice. In this case, the natural sample space is:

$$\Omega = \{HH, HT, TH, TT\}.$$

Remember we were asking what the probability distribution is, say $\mathbb{P}(HH)$. We can now construct the probability distribution in a rigorous manner. Let A and B be the events that the 1st and 2nd flips show "Head", respectively. Then, we can represent $\mathbb{P}(HH)$ as:

$$\begin{aligned}\mathbb{P}(HH) &= \mathbb{P}(A \cap B)\\ &= \mathbb{P}(A)\mathbb{P}(B)\\ &= \frac{2}{3} \times \frac{2}{3}\end{aligned}$$

where the second equality is due to the reasonable assumption that two flips are *independent*.

Example #2: Monty Hall Problem

The second example is the one that we studied in Lecture 3: the Monty Hall Problem. See Fig. ??. In Lecture 3, we considered the sample space in which each element takes a triplet:

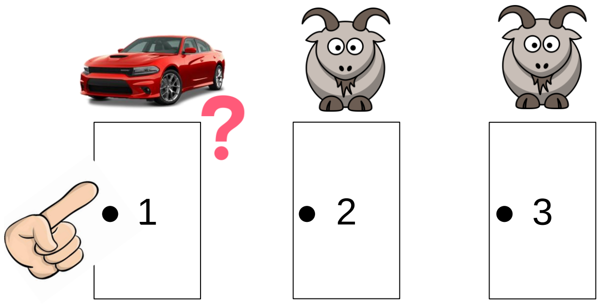


Figure 2: The setting of the Monty Hall Problem. The prize "car" is hidden behind a door, which is unknown to the trader. There are goats (sort of "qquang") behind the other two doors. The host knows about this setting. The trader is asked to pick a door for the prize "car".

(car's location, trader's choice, host's choice). Here we construct another sample space in which only two uncertainties are taken into consideration: car's location and trader's choice:

$$\Omega = \{(1, 1), (1, 2), (1, 3), \ldots, (2, 3), (3, 3)\}.$$

In this case, what is the probability distribution? It is obviously uniform. There are 9 cases and each case is not particularly different from the others due to symmetry. Hence, it should be uniform. We can also rigorously prove that it is indeed uniform, using the concept of independence. Let A be the event that car is located behind door i, and B the event that trader's choice is door j. Then, we can rewrite $\mathbb{P}((i,j))$ as:

$$\begin{aligned}\mathbb{P}((i,j)) &= \mathbb{P}(A\cap B)\\ &= \mathbb{P}(A)\mathbb{P}(B)\\ &= \frac{1}{3}\times\frac{1}{3}\end{aligned} \tag{17}$$

where the 2nd equality follows from the independence of car's location and trader's choice.

Example #3: Balls & Bins

The third example is the one which we did not explore, but which is very famous. That is a problem, so called the *Balls-&-Bins* problem. See Fig. **??**. There are k balls and n bins (“baguni” in Korean). We throw each ball into one of the n bins, uniformly at random. This throwing is done independently from the other remaining balls.

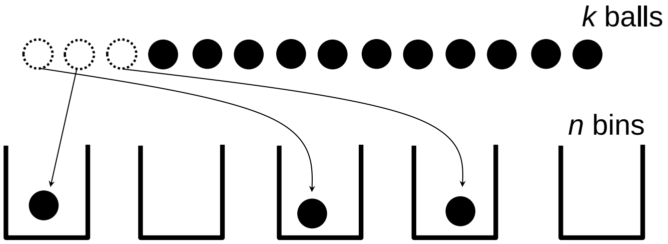


Figure 3: Balls-&-Bins Problem: Throw each ball into a bin, uniformly at random. Repeat this independently from other balls.

Suppose we are interested in figuring out the probability that the 1st bin is still empty even after throwing all of the k balls. Let A_i be the event that the ith ball is not placed in bin 1. Using these notations, we can then compute the probability as:

$$\begin{aligned}\mathbb{P}(\text{1st bin empty}) &= \mathbb{P}(A_1\cap A_2\cap\cdots\cap A_k)\\ &= \mathbb{P}(A_1)\cap\mathbb{P}(A_2)\cap\cdots\cap\mathbb{P}(A_k)\\ &= \mathbb{P}(A_1)\mathbb{P}(A_2)\cdots\mathbb{P}(A_k)\\ &= \left(\frac{n-1}{n}\right)^k\end{aligned} \tag{18}$$

where the 2nd equality is due to the mutual independence of $(A_1,\ldots,A_k)$; and the last equality is due to $\mathbb{P}(A_i)=\frac{n-1}{n}\ \forall i\in\{1,2,\ldots,k\}$.

Example #4: Fair vs biased coins

The last example is a bit tricky one. See Fig. **??**. There are two coins. One is fair and the other is biased with probability p of showing “Head”. We consider two experiments. In Experiment 1, we randomly choose a coin between the two and then flip the coin once. We repeat such a procedure one more time in an independent manner. An interested question is: What is $\mathbb{P}(HH)$? To figure this out, let A_i be the event that the ith flip is “Head”, and B_i the event that the fair coin is chosen in the ith flip. We then can rewrite $\mathbb{P}(HH)$ as:

$$\begin{aligned}\mathbb{P}(HH) &= \mathbb{P}(A_1\cap A_2)\\ &= \mathbb{P}(A_1)\mathbb{P}(A_2)\end{aligned} \tag{19}$$

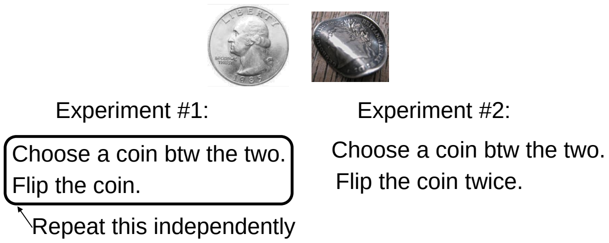

Experiment #1:
Choose a coin btw the two.
Flip the coin.
Repeat this independently
Experiment #2:
Choose a coin btw the two.
Flip the coin twice.

Figure 4: Fair-vs-biased coins: There are two coins: one is fair and the other is biased with the "Head" probability is p; (Experiment 1): We randomly choose a coin between the two and then flip the coin. We repeat this independently; (Experiment 2): We randomly choose a coin between the two. We then flip the same coin twice.

where the last equality is because two trials in Experiment 1 are independent with each other. We focus on one probability:

$$\begin{aligned}\mathbb{P}(A_1) &= \mathbb{P}(B_1 \cap A_1) + \mathbb{P}(B_1^c \cap A_1)\\ &= \mathbb{P}(B_1)\mathbb{P}(A_1|B_1) + \mathbb{P}(B_1^c)\mathbb{P}(A_1|B_1^c)\\ &= \frac{1}{2}\left(\frac{1}{2}+p\right)\end{aligned}$$

where the 1st equality is due to the total probability law and the 2nd follows from the definition of conditional probability. Plugging this into (**??**) and apply the symmetry argument w.r.t. $\mathbb{P}(A_2)$, we get:

$$\mathbb{P}(HH) = \mathbb{P}(A_1 \cap A_2) = \frac{1}{4}\left(\frac{1}{2}+p\right)^2. \tag{20}$$

Now consider Experiment 2. Here the procedure is slightly different. First, we randomly choose a coin between the two. We then flip the same chosen coin *twice*. We ask the same question: What is $\mathbb{P}(HH)$? We know that

$$\mathbb{P}(HH) = \mathbb{P}(A_1 \cap A_2). \tag{21}$$

One natural question is then: are A_i's independent? It turns out it is not the case. So we need to compute the interested probability in a different manner. Here one key event that we need to introduce is the one, say B, that the initially chosen coin is fair. Using this together with the total probability law, we can then obtain:

$$\begin{aligned}\mathbb{P}(HH) &= \mathbb{P}(A_1 \cap A_2)\\ &= \mathbb{P}(B \cap A_1 \cap A_2) + \mathbb{P}(B^c \cap A_1 \cap A_2)\\ &= \mathbb{P}(B)\mathbb{P}(A_1 \cap A_2|B) + \mathbb{P}(B^c)\mathbb{P}(A_1 \cap A_2|B^c)\\ &= \frac{1}{2}\left(\frac{1}{4}+p^2\right)\end{aligned} \tag{22}$$

where the 2nd and 3rd equalities follow from the total probability law and the definition of conditional probability, respectively. Here the last step is the key. Once a coin is chosen, say given the event B (the initially chosen coin is fair), flipping the coin in the first is independent of the second trial. Hence, given B, A_1 and A_2 are independent:

$$\mathbb{P}(A_1 \cap A_2|B) = \mathbb{P}(A_1|B)\mathbb{P}(A_2|B) = \frac{1}{2}\times\frac{1}{2} = \frac{1}{4}.$$

Similarly,

$$\mathbb{P}(A_1 \cap A_2 | B^c) = \mathbb{P}(A_1|B^c)\mathbb{P}(A_2|B^c) = p \times p = p^2.$$

So we obtain the last equality in (**??**). Here one thing that you may notice is that *two dependent events can be conditionally independent.* The formal definition of *conditional independence* is: We say that events A and B are conditionally independent w.r.t. C if

$$\mathbb{P}(A \cap B|C) = \mathbb{P}(A|C)\mathbb{P}(B|C). \tag{23}$$

Also, there are many cases in which *two independent events can be conditionally dependent.* One such example is the one that we investigated earlier as a counter-example in which events are pairwise independent but not mutually independent. Why? Think about it.

Finally let us verify what I claimed as above: the two events A_1 and A_2 are *dependent.* To see this, we compute:

$$\begin{aligned} \mathbb{P}(A_1) &= \mathbb{P}(B)\mathbb{P}(A_1|B) + \mathbb{P}(B^c)\mathbb{P}(A_1|B^c) \\ &= \frac{1}{2}\left(\frac{1}{2} + p\right) \end{aligned} \tag{24}$$

where the 1st equality comes again from the total probability law and the definition of conditional probability. Hence, we verify the dependence via:

$$\mathbb{P}(A_1 \cap A_2) = \frac{1}{2}\left(\frac{1}{4} + p^2\right) \neq \frac{1}{4}\left(\frac{1}{2} + p\right)^2 = \mathbb{P}(A_1)\mathbb{P}(A_2).$$

Look ahead

So far we have exercised ourselves with somewhat easy examples. So you may be very confident about the concept of independence. As a matter of fact, the independence concept is very deep and tricky. There are tons of non-trivial examples where we can answer interested questions only via smart handling of the concept. Next time, we will investigate one such example, called the *coupon collector problem.*

EE210 Probability and Introductory Random Processes March 18, 2021
KAIST, Spring 2021 Changho Suh (chsuh@kaist.ac.kr)

Lecture 6: Coupon Collector Problem

Recap

Last time, we learned about an important concept: *independence*. For a simple setting where there are two interested events, the definition was simple and intuitive particularly with the help of the definition of conditional probability. We say that two events A and B are independent if

$$\mathbb{P}(A \cap B) = \mathbb{P}(A)\mathbb{P}(B). \tag{1}$$

But the general case in which there are an arbitrary number of events involved was tricky. There are many types of "something" independence: *mutual* independence, *pairwise* independence, and more generally *k-way* independence. There are tons of cases in which one type of independence holds while the other does not. Given this chaos, my recommendation was to *simply memorize the definition of mutual independence* and then start from there. We say that events $(A_1, \ldots, A_n)$ are mutually independent if

$$\mathbb{P}\left(\bigcap_{i\in I} A_i\right) = \prod_{i\in I} \mathbb{P}(A_i) \quad \forall I \subseteq \{1, 2, \ldots, n\}. \tag{2}$$

We also exercised ourselves on the independence concept with several examples which are not that difficult. At the end of the last lecture, however, I claimed that there are many non-trivial examples in which the independence plays a crucial role, yet it is non-straightforward to apply though.

Today's lecture

Today we will study one such example in depth: the *Coupon Collector Problem*. This lecture consists of four parts. In the first part, I will explain what the problem is and introduce an interesting question that arose in the problem context. It turns out directly addressing the question is difficult, but there is a simpler version of the question that can give some insights. So in the second part, we will investigate the simpler version particularly with the help of the concept of independence. Next we will study one important technique that serves to address the original difficult question together with the solution of the simpler version. Lastly we will employ the technique to come up with a solution to the original question.

Coupon Collector Problem

The problem that we will study is a very famous one, called the *Coupon Collector Problem*. Here is the setting of the problem. Consider a snack that contains a coupon inside. One such snack is "Cheetos" that you might enjoy like my son. In the case of Cheetos, the coupon looks like the one in Fig. 1 – it is like "ddack-ji" in Korean. Suppose that there are n different kinds of coupons, and each snack has a coupon chosen from n possibilities uniformly at random, independently of the other snacks. One very prominent question that was raised in this problem context is:

How many snacks need to buy to have at least a 90% chance of collecting all the coupons?

Figure 1: The coupon collector problem: There are n different coupons. A coupon is chosen uniformly at random (out of n) to be contained in a snack ("Cheetos" in this example). Suppose we buy k snacks.

This question can be formally written as follows. Let k be the number of snacks purchased. What is k such that

$$\mathbb{P}(\text{obtain every coupon from } k \text{ snanks}) \geq 0.9? \tag{3}$$

As mentioned in the beginning, directly addressing the question is not that easy. So let's introduce a simpler version of the question that turns out to shed light.

A simpler version of the question (3)

The simpler version is concerned about obtaining only one interested coupon, say Coupon 1. So the corresponding question reads: What is k such that

$$\mathbb{P}(\text{obtain "Coupon 1" from } k \text{ snanks}) \geq 0.9? \tag{4}$$

It turns out one can readily answer this question using the concept of independence. The important step for the use of the independence concept is to smartly introduce multiple events so that they are *independent*. Inspired by the fact that each snack having a coupon is independent of the other snacks, one can think of the following events. Let A_i be the event that the ith snack contains "Coupon 1". These events are then independent due to our underlying assumption. Then, the above probability (the left-hand-side in (4)) can be written as:

$$\begin{aligned}&\mathbb{P}(\text{obtain "Coupon 1" from } k \text{ snanks})\\&= \mathbb{P}(A_1 \cup A_2 \cup \cdots \cup A_k).\end{aligned} \tag{5}$$

But here comes an issue. The issue is that the interested event is expressed in terms of the *union* of the events A_i's. Hence, this prevents us from exploiting the independence property: $\mathbb{P}(A_1 \cap A_2 \cap \cdots \cap A_k)$. But there is a trick which allows us to resolve the issue. The trick is the one that we learned in Lecture 2. It is the *complement* trick. Here the complement trick together one important law that you learned (De Morgan's law) comes to rescue. Applying the De Morgan's law ($(A_1 \cup A_2)^c = A_1^c \cap A_2^c)$) into (5) $k-1$ times, we get:

$$\begin{aligned}&\mathbb{P}(\text{obtain "Coupon 1" from } k \text{ snanks})\\&= \mathbb{P}(A_1 \cup A_2 \cup \cdots \cup A_k)\\&= 1 - \mathbb{P}(A_1^c \cap A_2^c \cap \cdots \cap A_k^c)\end{aligned} \tag{6}$$

Here one key observation is that the events $(A_1^c, \ldots, A_k^c)$ are also mutually independent. This is again because of our assumption that each snack contains a coupon uniformly at random, *independently* of the other snacks. Applying the independence property, we get:

$$\begin{aligned}
&\mathbb{P}(\text{obtain “Coupon 1” from } k \text{ snanks}) \\
&\quad = 1 - \mathbb{P}(A_1^c \cap A_2^c \cap \cdots \cap A_k^c) \\
&\quad = 1 - \mathbb{P}(A_1^c) \cdots \mathbb{P}(A_k^c) \qquad (7) \\
&\quad = 1 - \left(1 - \frac{1}{n}\right)^k
\end{aligned}$$

where the last equality follows from $\mathbb{P}(A_i^c) = \frac{n-1}{n}$ $\forall i \in \{1, 2, \ldots, k\}$. Putting this into (4) together with a proper massaging, we get:

$$k \log\left(1 - \frac{1}{n}\right) \le \log 0.1$$

Since $\log\left(1 - \frac{1}{n}\right) < 0$, diving both sides by $\log\left(1 - \frac{1}{n}\right)$ gives:

$$k \ge \frac{\log 0.1}{\log\left(1 - \frac{1}{n}\right)}. \qquad (8)$$

Focusing on one extreme case in which n is very large, we can simplify the expression (8) so that is looks a bit more intuitive. The simplification builds upon an approximation based on Taylor's series. Let $f(x) = \log(1 + x)$. Here log indicates log to the base e. Then, for a very small x,

$$\log(1 + x) \approx f(0) + \frac{f'(0)}{1!}x = x, \qquad (9)$$

as $f'(0) = \frac{1}{1+x}\Big|_{x=0} = 1$. Applying this into $\log\left(1 - \frac{1}{n}\right)$ for a very large n, we can map x to $-\frac{1}{n}$, thus yielding:

$$\log\left(1 - \frac{1}{n}\right) \approx -\frac{1}{n}.$$

Apply this approximation into (8), we get:

$$k \ge \frac{\log 0.1}{\log\left(1 - \frac{1}{n}\right)} \approx n \log 10.$$

This result means that in order to obtain one coupon of interest, we need to buy snacks as many as the number that grows *linearly* in n.

Go back to the original question (3)

Now let us go back to the original question which concerns the following probability:

$$\mathbb{P}(\underbrace{\text{obtain every coupon from } k \text{ snanks}}_{=:E}).$$

Letting by E the interested event, our goal is to compute $\mathbb{P}(E)$. In an effort to build upon the simpler version, denote by E_i the event that we obtain “Coupon i” from k snacks. Then, the interested event E can be written as $E = E_1 \cap E_2 \cap \cdots \cap E_n$. Hence, we get:

$$\mathbb{P}(E) = \mathbb{P}(E_1 \cap E_2 \cap \cdots \cap E_n). \qquad (10)$$

In (7), $\mathbb{P}(E_1) = 1 - \left(1 - \frac{1}{n}\right)^k$. By symmetry, $\mathbb{P}(E_1) = \cdots = \mathbb{P}(E_n)$. This then gives:

$$\mathbb{P}(E_i) = 1 - \left(1 - \frac{1}{n}\right)^k. \tag{11}$$

Gazing at (10) and (11), you may be highly tempted to believe that $(E_1, \ldots, E_n)$ are mutually independent. Unfortunately, it turns out this is not the case. Here is a counterexample for a simple setting $(n, k) = (2, 2)$. In this case,

$$\mathbb{P}(E_i) = 1 - \left(1 - \frac{1}{2}\right)^2 = \frac{3}{4} \quad\longrightarrow\quad \mathbb{P}(E_1)\mathbb{P}(E_2) = \frac{9}{16}. \tag{12}$$

On the other hand,

$$\begin{aligned} \mathbb{P}(E_1 \cap E_2) &= \mathbb{P}(\text{obtain every coupon from two snacks}) \\ &= \mathbb{P}(\text{two coupons in the two snacks are different}) \\ &= \frac{2}{4} = \frac{1}{2} \end{aligned} \tag{13}$$

where the second last equality comes from the fact that there are two coupon configurations for the desired event (marked in blue below) out of 4:

$$\{(\text{Coupon } 1, \text{Coupon } 1), (\text{Coupon } 1, \text{Coupon } 2), (\text{Coupon } 2, \text{Coupon } 1), (\text{Coupon } 2, \text{Coupon } 2)\}.$$

With (12) and (13), we check the dependence of (E_1, E_2):

$$\mathbb{P}(E_1 \cap E_2) = \frac{1}{2} \neq \frac{9}{16} = \mathbb{P}(E_1)\mathbb{P}(E_2). \tag{14}$$

Relax the goal

Now what can we do then in the difficult situation that comes from the dependence of $(E_1, \ldots, E_n)$ in (10)? Actually, there is one important work-around that we can take whenever we encounter a situation where the computation of an interested probability is difficult. That is to *relax* the situation. What it means by relaxing in the context of probability is to *derive a lower or upper bound on the interested probability* instead of targeting the *exact* probability. To figure out what this means in our problem setting, let's recall the original question raised. What is k such that

$$\mathbb{P}(\text{obtain every coupon from } k \text{ snanks}) = \mathbb{P}(E) \geq 0.9? \tag{15}$$

Consider a lower bound, say $\mathcal{L}_{\sf bound}$, on $\mathbb{P}(E)$: $\mathbb{P}(E) \geq \mathcal{L}_{\sf bound}$. Here one key observation that we can make is: whenever the lower bound is above the target chance 0.9, so is the exact probability:

$$\mathcal{L}_{\sf bound} \geq 0.9 \quad\Longrightarrow\quad \mathbb{P}(E) \geq 0.9. \tag{16}$$

This observation can lead us to set up the following relaxed goal: Finding k such that

$$\mathbb{P}(E) \geq \mathcal{L}_{\sf bound} \geq 0.9. \tag{17}$$

Under this relaxed goal, we are then interested in computing $\mathcal{L}_{\sf bound}$ instead. It turns out there is an important bounding technique that leads us to compute $\mathcal{L}_{\sf bound}$: the *union bound*.

Union bound

The union bound is very simple to state. For two events, say A and B, it says:

$$\mathbb{P}(A \cup B) \leq \mathbb{P}(A) + \mathbb{P}(B). \tag{18}$$

The proof of this is straightforward. Using the definition of an event,

$$\mathbb{P}(A \cup B) = \mathbb{P}(A) + \mathbb{P}(B) - \mathbb{P}(A \cap B). \tag{19}$$

Why? Think about a Venn diagram. Since $\mathbb{P}(A \cap B) \geq 0$, we get (18).
Now how to apply the union bound into the interested probability:

$$\mathbb{P}(E) = \mathbb{P}(E_1 \cap E_2 \cap \cdots \cap E_n)? \tag{20}$$

To this end, using De Morgan's law, we first get:

$$\mathbb{P}(E^c) = \mathbb{P}(E_1^c \cup E_2^c \cup \cdots \cup E_n^c). \tag{21}$$

Applying the union bound (18) into the above multiple times (precisely $n-1$ times), we get:

$$\begin{aligned}
\mathbb{P}(E^c) &= \mathbb{P}(E_1^c \cup E_2^c \cup \cdots \cup E_n^c) \\
&\leq \mathbb{P}(E_1^c) + \mathbb{P}(E_2^c \cup \cdots \cup E_n^c) \\
&\ \ \vdots \\
&\leq \mathbb{P}(E_1^c) + \mathbb{P}(E_2^c) + \cdots + \mathbb{P}(E_n^c) \\
&= n\left(1 - \frac{1}{n}\right)^k
\end{aligned} \tag{22}$$

where the last equality follows from $\mathbb{P}(E_i^c) = \left(1 - \frac{1}{n}\right)^k \ \forall i \in \{1, \ldots, n\}$; see (11). Hence, we get:

$$\mathbb{P}(E) = 1 - \mathbb{P}(E^c) \geq 1 - n\left(1 - \frac{1}{n}\right)^k =: \mathcal{L}_{\sf bound}. \tag{23}$$

How many snacks do we need to buy under the relaxed goal?

Now let's figure out the number of snacks required to complete all the coupons under the relaxed goal targeting:

$$\mathbb{P}(E) \geq \mathcal{L}_{\sf bound} \geq 0.9. \tag{24}$$

Putting (23) into the above, we get:

$$k \log\left(1 - \frac{1}{n}\right) \leq \log \frac{0.1}{n}.$$

Dividing $\log\left(1 - \frac{1}{n}\right) < 0$ on both sides,

$$k \geq \frac{\log \frac{0.1}{n}}{\log\left(1 - \frac{1}{n}\right)}. \tag{25}$$

Again, focusing on one extreme case in which n is very large, we obtain:

$$\log\left(1 - \frac{1}{n}\right) \approx -\frac{1}{n}$$

using Taylor's series approximation. Apply this into (25), we get:

$$k \geq \frac{\log \frac{0.1}{n}}{\log \left(1 - \frac{1}{n}\right)} \approx n \log(10n).$$

What this results means is that in order to collect all of the n coupons, we need to buy many snacks so that the number grows *super-linearly* in n.

Look ahead

We have thus far studied many important concepts, laws and techniques: sample space, probability model, probability distribution, events, conditional probability, total probability law, Bayes' law, independence and union bound. Unfortunately, there are several more concepts that we need to figure out in order to understand the MAP and ML estimation principles that we aimed at learning in the first lecture. Next time, we will investigate one of the remaining concepts: *random variables*.

EE210 Probability and Introductory Random Processes March 23, 2021
KAIST, Spring 2021 Changho Suh (chsuh@kaist.ac.kr)

Lecture 7: Random variables

Recap

During the past lectures, we have explored numerous concepts, laws and techniques: sample space, probability model, probability distribution, events, the complement counting trick, conditional probability, total probability law, Bayes' law, independence, De Morgan's law and union bound. I believe you have been very much familiar with those particularly with the help of a couple of non-trivial examples such as the birthday paradox problem, the Monty Hall problem, the disease testing problem, and the coupon collector problem. While you may now feel confident about the probability, this is not the end of the story in light of the goal that we set out in Lecture 1: understanding the two key principles: the MAP and ML estimation principles. At the end of the last lecture, I mentioned that in order to figure these out, we need to learn about two more concepts: (i) *random variables*; and (ii) *random processes*.

Today's lecture

Today we will investigate one of them: Random variables. Specifically what we are going to do are five folded. First we will introduce the definition of a random variable. It turns out there is a *probability model* associated with a random variable. So we will next study the corresponding probability model. As you may imagine, similar to events, there is a concept of the *independence* w.r.t. random variables. In the 3rd part, we will introduce the definition of independence tailored for random variables. As we will figure out soon, a random variable can also be expressed in terms of other more basic random variables. So in the 4th part, we will investigate a function of random variables together with its corresponding probability model. In particular, we will focus on one specific yet popular function: *summation*. Lastly, as usual, we will exercise ourselves on the learned concepts with some examples.

Definition of a random variable

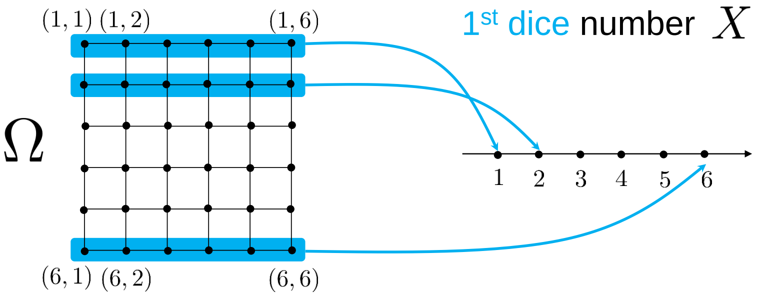


Figure 1: One example of a random variable: the 1st dice number in an experiment of rolling two fair dice with a sample space $\Omega = \{(1,1),(1,2),\ldots,(5,6),(6,6)\}$. The random variable, say X, is simply a function that outputs a *real-valued* number fed by an outcome in Ω. For instance, $X = 1$ is mapped from 6 outcomes with the 1st dice number being 1: $(1,1),(1,2),\ldots,(1,6)$.

A random variable is a *function* that maps something to another. Here the "something" (formally referred to as the *domain*) is an *outcome* in a sample space Ω, and the "another" (formally referred to as the *range*) indicates a *real-valued number*. So it is nothing but a *transformation* that converts outcomes (not necessarily numerical values) into numbers. In order to get a concrete feel as to what it means, let's consider one of the experiments that we investigated earlier: Rolling two fair dice. See Fig. 1. Here the sample space consists of a bunch of pairs (ω_1, ω_2) where ω_1 and ω_2 denote the 1st and 2nd dice numbers, respectively. We have 6×6 elements and these are visualized as dots in the two-dimensional picture; see the left figure for illustration. In this example, we consider a *random variable* that takes the 1st dice number ω_1 from an outcome $\omega = (\omega_1, \omega_2)$. Usually the random variable is denoted by a capital letter, so let's say X. To emphasize it is a *function* of an outcome ω, some people denote it by $X(\omega)$, but many people prefer to use a simpler expression X, as it can easily be figured out from the context. As illustrated via cyan-blue arrows, the random variable X is a mapping that yields the 1st dice number from outcome pairs (ω_1, ω_2). For instance, $X(\omega) = 1$ (simply denoted by $X = 1$) is mapped from six outcomes: $(1, 1), (1, 2), \ldots, (1, 6)$.

Since a random variable is a *function*, you can easily image that there could be many random variables. Yes, we can construct many. Another natural random variable, say Y, that you can think of is a function that outputs the 2nd dice number from an outcome. Or a bit twisted random variable, say S, is the one that yields the sum of the two dice number. See Fig. 2. In this case, a particular value of S, say $S(\omega) = 4$ ($S = 4$) is mapped from three outcomes: $(1, 3), (2, 2), (3, 1)$, marked in a green diagonal rounded-square.

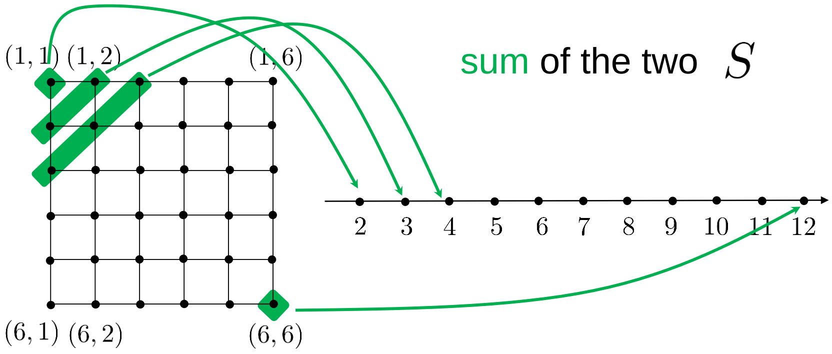


Figure 2: Another random variable S: a function that yields the sum of the two dice number. The event $S = s$ is mapped from multiple outcomes subject to $\omega_1 + \omega_2 = s$ where $(\omega_1, \omega_2) \in \Omega$. For instance, $S = 4$ is associated with three outcomes: $(1, 3), (2, 2), (3, 1)$.

Rationale behind the definition

Now you may wonder why we define random variables. In other words, why do we care about such from-outcomes-to-numbers mappings? The reason is that we are *often interested in numerical values* w.r.t outcomes. Rolling two dice is just one such experiment. There are many more experiments that concern numerical values. For instance, we may be interested in: (i) the number of "Head"s in an experiment of flipping a coin n times; (ii) the number of students who have the same birthday in the birthday paradox problem; or (iii) the number of collected coupons from k snacks bought in the coupon collector problem. These are merely a few instances. As you may imagine, there are tons of examples.

Probability model of a random variable

In Lecture 2, we learned about one important concept: the probability model that consists of a sample space and its corresponding probability distribution. In the context of random variables, the same thing happens. We also have a probability model w.r.t. a random variable. As before, the probability model is composed of two entities. The first is the set of values that an interested random variable, say X, can take on. It is usually denoted by a caligraphic version $\mathcal{X}$ of the letter X used to denote the random variable:

$$\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$$

where x_i indicates a numerical value that X can take on as one particular realization. By convention, we use a small letter to indicate a certain realization. The set $\mathcal{X}$ is simply called the *range*.

The second entity is the probability distribution. Relative to the probability distribution w.r.t. the sample space $\mathbb{P}(\omega)$, a slightly different notation is used: $\mathbb{P}_{X}(x)\ \forall x \in \mathcal{X}$. This is because we want to differentiate it from $\mathbb{P}(\omega)$. We place the random variable X in the subscript. There is another name for the probability distribution: the probability mass function, "pmf" for short. Its computation can be done exactly in the same manner as we do for an event. This is because $X = x$ can be seen as a particular event:

$$\mathbb{P}_X(x) = \mathbb{P}(X = x) = \sum_{\omega \in \Omega : \omega \xrightarrow{X} x} \mathbb{P}(\omega). \tag{1}$$

Here "$\omega \in \Omega : \omega \xrightarrow{X} x$" placed below in the summation means "over all ω's such that ω yields x via the mapping X. Usually in mathematics, the symbol colon ":" means "such that" or "subject to".

In the above rolling-two-dice experiment, we can readily construct the probability model. For X (1st dice number), the range reads $\mathcal{X} = \{1, 2, \ldots, 6\}$, and its probability distribution is

$$\mathbb{P}_X(x) := \sum_{\omega=(\omega_1,\omega_2)\in\Omega:\ \omega_1 = x} \mathbb{P}(\omega) = \frac{6}{36} = \frac{1}{6}.$$

Again, "$\omega = (\omega_1, \omega_2) \in \Omega :\ \ \omega_1 = x$" placed below in the summation means "over all ω's subject to $\omega_1 = x$". For another random variable S, on the other hand, the range reads: $\mathcal{S} = \{2, 3, \ldots, 12\}$. The corresponding probability distribution can be computed as follows: for instance, when $s = 4$,

$$\mathbb{P}_S(4) := \sum_{\omega=(\omega_1,\omega_2)\in\Omega:\ \omega_1+\omega_2=4} \mathbb{P}(\omega) = \frac{3}{36}.$$

One key property of the probability distribution

There is one key property that the probability distribution should respect. The property comes from two observations that hold for any random variables. For illustrative purpose, let us explain them via the random variable S. The first is that any two events, say $S = s_1$ and $S = s_2$, are *disjoint* whenever $s_1 \neq s_2$. It is because the two events never occur at the same time for $s_1 \neq s_2$. This can also be understood by the property of a *function* which a random variable belongs to: a single input of a function cannot yield two different outputs, although the other way around holds, i.e., one output can be mapped from many distinct inputs. The second observation is that the union of all the disjoint events covers the entire sample space Ω. For every outcome $\omega \in \Omega$, there is a mapping to a particular event; hence, all of the possible events should span all the elements in Ω. These two observations are illustrated in Fig. 3. These two observations

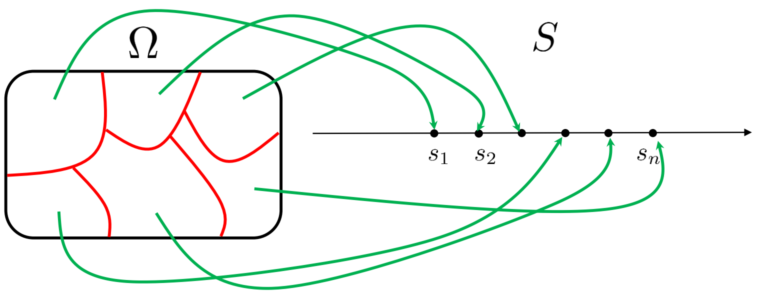


Figure 3: Two observations that hold for any random variables: (i) the events $S = s$ are *disjoint* for different s's; (ii) for every outcome $\omega \in \Omega$, there is a mapping to a particular s, and therefore all of the possible events should span all the elements in Ω.

then yield:

$$\begin{aligned}\sum_{s \in \mathcal{S}} \mathbb{P}_S(s) &= \sum_{s \in \mathcal{S}} \mathbb{P}(S = s) \\ &= \sum_{s \in \mathcal{S}} \sum_{\omega \in \Omega : S(\omega) = s} \mathbb{P}(\omega) \\ &= \sum_{\omega \in \Omega} \mathbb{P}(\omega) = 1\end{aligned} \tag{2}$$

where the last equality follows from the fact that the events $S = s$ are disjoint for different s's (1st observation) and the union of all the disjoint events spans Ω (2nd observation). Similar to the sum-up-to-one constraint in the sample space, you can view this as a sort of its random-variable counterpart.

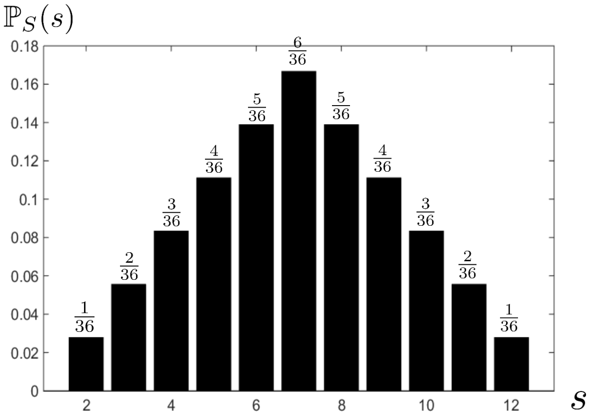


Figure 4: Histogram visualization of the probability distribution of the random variable S that indicates the sum of the two dice number in an experiment of rolling two fair dice.

The probability distribution is often visualized via histogram. For instance, $\mathbb{P}_S(s)$ looks like Fig. 4. Here the height of each bar denotes its associated probability. From this, we can also check that the sum of $\mathbb{P}_S(s)$ for all s's is indeed 1: $\frac{1}{36}(1 + 2 + \cdots + 5 + 6 + 5 + \cdots + 2 + 1) = \frac{21+15}{36} = 1$.

Independence of random variables

Similar to events, we have the independence concept for random variables. For *two* random variables, say X and Y, X and Y are said to be *independent* if

$$\text{any two events } \{X = x\} \text{ and } \{Y = y\} \text{ are independent } \; \forall x \in \mathcal{X},\ \forall y \in \mathcal{Y}. \tag{3}$$

In the general case with an arbitrary number of random variables, we say that $(X_1, \ldots, X_n)$ are *mutually independent* if

$$(\{X_1 = x_1\}, \ldots, \{X_n = x_n\}) \text{ are mutually independent } \; \forall x_1 \in \mathcal{X}_1, \ldots, \forall x_n \in \mathcal{X}_n. \tag{4}$$

A function of random variables

Since a random variable is a function, one can also think of a function of the function, i.e., a *composite* function. The composite function can be interpreted as a function of sort of basic random variables. Obviously it is also a random variable. As the most prominent and useful function, people often consider "summation". So in this section, we explore something relevant to the sum of random variables.

To be concrete, we will do this exploration via the previous example regarding S (the sum of the two dice number in the rolling-two-dice experiment). As you can readily imagine, S can also be represented in terms of more basic random variables X and Y (the 1st and 2nd dice numbers):

$$S = X + Y.$$

So it is a perfect example that concerns the sum of random variables.

One thing that I would like to put a special emphasis on is about a way to compute $\mathbb{P}_S(s)$. Remember in the previous section that we computed $\mathbb{P}_S(s)$ via counting the number of cases ω's that yield $\omega_1 + \omega_2 = s$. It turns out there is another way for such computation using the probability distributions of the component random variables $\mathbb{P}_X(x)$ and $\mathbb{P}_Y(y)$. Here is the way. We first obtain:

$$\mathbb{P}_S(s) = \mathbb{P}(X + Y = s).$$

This is because $S = X + Y = s$ can be seen as an event. Here one key observation that we can make is that there are two sources of uncertainties in the event $X + Y = s$: one is w.r.t. X; the other is w.r.t. Y. Due to multiple uncertainties, there are many subcases that yield the event $X + Y = s$. There is one very important law (that we learned), which plays a crucial role in such many-subcases scenario. That is, the *total probability law.* So using the TPL, we then write:

$$\begin{aligned}\mathbb{P}_S(s) &= \mathbb{P}(X + Y = s) \\ &= \sum_{x=1}^{s-1} \mathbb{P}(\{X = x\} \cap \{Y = s - x\}).\end{aligned}$$

Here the expression that includes "$\cap$" is a bit dirty. So people often use the following simpler expression: $\mathbb{P}(\{X = x\} \cap \{Y = s - x\}) = \mathbb{P}(X = x, Y = s - x)$, where the symbol comma "," means "and". Adopting this simpler expression, we get:

$$\begin{aligned}\mathbb{P}_S(s) &= \mathbb{P}(X + Y = s) \\ &= \sum_{x=1}^{s-1} \mathbb{P}(X = x, Y = s - x) \\ &= \sum_{x=1}^{s-1} \underbrace{\mathbb{P}(X = x)}_{\mathbb{P}_X(x)} \underbrace{\mathbb{P}(Y = s - x)}_{\mathbb{P}_Y(s-x)}\end{aligned}$$

where the last equality comes from the fact that the events $X = x$ and $Y = s-x$ are independent (i.e., X and Y are independent). Recognizing $\mathbb{P}(X = x)$ and $\mathbb{P}(Y = s-x)$ as $\mathbb{P}_X(x)$ and $\mathbb{P}_Y(s-x)$ respectively, we finally obtain:

$$\mathbb{P}_S(s) = \sum_{x=1}^{s-1} \mathbb{P}_X(x)\mathbb{P}_Y(s-x). \tag{5}$$

Here the complicated-looking expression in the right-hand-side is actually the very famous operation that some of you may be familiar with. That is, *convolution*, denoted by $(\mathbb{P}_X * \mathbb{P}_Y)(s)$. The convolution is a prominent operation that is particularly powerful in the field of Electrical Engineering. So it is covered in several basic courses offered in the department of Electrical Engineering. One such basic course is “EE202 Signals & Systems”. So the students who are taking this course may be learning about the convolution. The formal definition is given by

$$(\mathbb{P}_X * \mathbb{P}_Y)(s) := \sum_{x=0}^{s} \mathbb{P}_X(x)\mathbb{P}_Y(s-x)$$

where the starting point is 0 and the end point is s (the interested point). This does not exactly match with the right-hand-side in (5). But these are actually matching. This is because the two points $x = 0$ and $x = s$ do not contribute to the summation, as $\mathbb{P}_X(0) = 0$ and $\mathbb{P}_Y(0) = 0$. Hence, we get:

$$\begin{aligned}\mathbb{P}_S(s) &= \sum_{x=1}^{s-1} \mathbb{P}_X(x)\mathbb{P}_Y(s-x)\\ &= \sum_{x=0}^{s} \mathbb{P}_X(x)\mathbb{P}_Y(s-x)\\ &=: (\mathbb{P}_X * \mathbb{P}_Y)(s).\end{aligned} \tag{6}$$

Actually, the convolution operation appears in the case where X and Y are independent. If they are *dependent*, (6) does not hold any more. In this case, we need to think about another way to compute $\mathbb{P}_S(s)$, which may differ depending on scenarios.

Example #1: Tossing a p-biased coin n times

Let us investigate two examples which are relevant to the sum of random variables. The first is a very simple example in which we toss a p-biased coin n times. Here the “p-biased” means that the probability of showing “Head” is p.

Suppose we are interested in the total number of “Head”s. Then, the number can be represented as the sum of the following basic random variables X_i's:

$$X_i = \begin{cases} 1, & \text{if the } i\text{th flips shows “Head”;}\\ 0, & \text{otherwise.}\end{cases}$$

Someone may prefer a shorthand notation: $X_i = \mathbf{1}\{i\text{th flips shows “Head”}\}$ where $\mathbf{1}\{\cdot\}$ is the indicator function that returns 1 when $(\cdot)$ is true while returning 0 otherwise. Using these, we write:

$$S = X_1 + X_2 + \cdots + X_n$$

where the range is $\mathcal{S} = \{0, 1, \ldots, n\}$.

Now how to compute the probability distribution $\mathbb{P}_S(s)$? Someone may wish to use the convolution operation (6) to proceed. But it turns out there is a much simpler way to go. The way is the one that exploits the *symmetry* property. Consider an event $S = s$. There are many configurations that yield $S = s$, but their corresponding probabilities are equal by *symmetry*. Hence,

$$\mathbb{P}_S(s) = \text{ (number of flip patterns yielding } s) \times \mathbb{P}(X_1 = 1, \ldots, X_s = 1, X_{s+1} = 0, \ldots, X_n = 0).$$

Here the number of flip patterns yielding s is $\binom{n}{s}$. The second probability can readily be computed using the independence (4) of $(X_1, \ldots, X_n)$:

$$\begin{aligned}\mathbb{P}(X_1 = 1, \ldots, X_s = 1, X_{s+1} = 0, \ldots, X_n = 0) &= \mathbb{P}(X_1 = 1) \cdots \mathbb{P}(X_s = 1)\mathbb{P}(X_{s+1} = 0) \cdots \mathbb{P}(X_n = 0) \\ &= p^s(1-p)^{n-s}.\end{aligned}$$

This together with the counting number $\binom{n}{s}$ then gives:

$$\mathbb{P}_S(s) = \binom{n}{s} p^s (1-p)^{n-s}, \qquad s \in \mathcal{S} = \{0, 1, \ldots, n\}.$$

Actually this is a very famous distribution, named the *Binomial* distribution. It is simply denoted by $S \sim \mathsf{Bin}(n, p)$ where the symbol "$\sim$" means "is distributed according to". Fig. 5 illustrates some pdf examples which show how the distribution looks like.

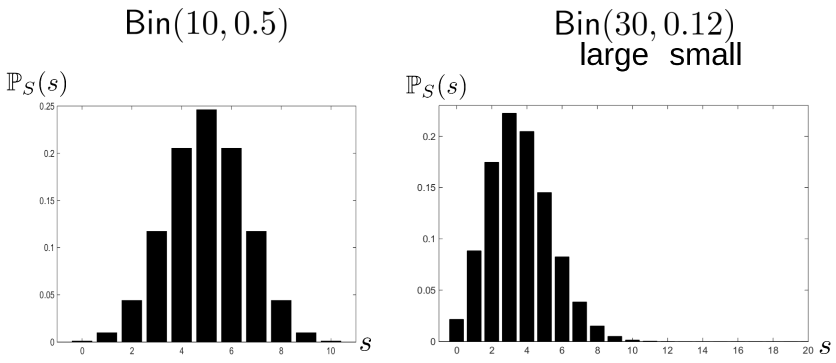


Figure 5: Histogram visualization of the binomial distribution. (Left): The case of $(n, p) = (10, 0.5)$; (Right): The case where n is large and p is small, e.g., $(n, p) = (30, 0.12)$.

Example #2: Homework matching

Consider another example in which an interested random variable can also be represented in terms of basic random variables, yet its distribution computation is not that simple. See Fig. 6. The homeworks of n students are collected in. The homeworks are randomly shuffled and then returned to students, so each student may not receive his/her own homework.

Suppose we are interested in the number of students who receive their own homeworks. Then, the number can be expressed as the sum of the following basic random variables X_i's:

$$\begin{aligned} X_i &= \begin{cases} 1, & \text{if the } i\text{th student receives her own homework;} \\ 0, & \text{otherwise.} \end{cases} \\ S &= X_1 + X_2 + \cdots + X_n. \end{aligned}$$

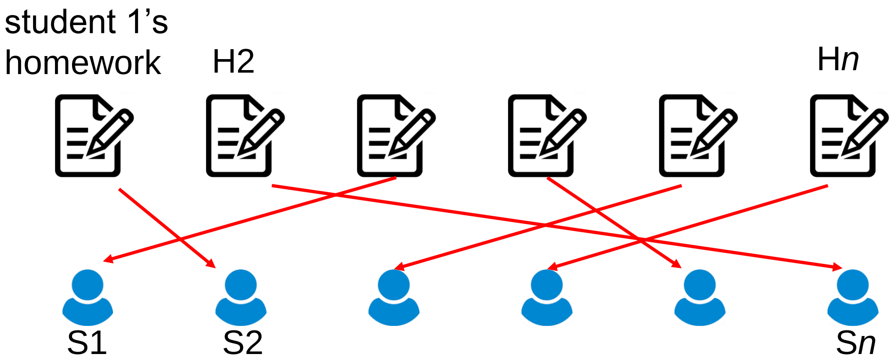


Figure 6: Homework matching problem: The homeworks collected from n students are shuffled and returned back to the students. Each student receives only one homework (one-to-one matching), yet the returned homework is not necessarily his/her own homework.

Similar to the prior example, we can use the *symmetry* property to obtain:

$$\mathbb{P}_S(s) = \ (\#\text{ of matching patterns yielding } s) \times \mathbb{P}(X_1 = 1, \ldots, X_s = 1, X_{s+1} = 0, \ldots, X_n = 0).$$

However, a challenge arises here. The challenge is two folded. The first is that the computation of the number of matching patterns yielding s is complicated. The second is a more serious one: the second probability quantity is difficult to compute. This is because X_i's are *dependent*. To see this, first compute $\mathbb{P}(X_1 = 1) = \frac{1}{n}$. But its conditional probability reads $\mathbb{P}(X_1 = 1|X_2 = 1) = \frac{1}{n-1}$, as the 2nd student receiving her own homework means there is one possibility for matching out of $n-1$ candidates. Hence, $\mathbb{P}(X_1 = 1) = \frac{1}{n} \neq \frac{1}{n-1} = \mathbb{P}(X_1 = 1|X_2 = 1)$. It turns out these challenges make the computation of $\mathbb{P}_S(s)$ quite difficult. So we will not attempt to do the computation.

Look ahead

In this lecture, we have focused on the probability model w.r.t. a *single* random variable. Of course, there is a probability model for *multiple* random variables. Next time, we will touch upon the content. We will also investigate one deterministic quantity that can somehow represent a random variable with uncertainty. That is, *expectation*.

EE210 Probability and Introductory Random Processes — March 25, 2021
KAIST, Spring 2021 — Changho Suh (chsuh@kaist.ac.kr)

Lecture 8: Joint probability distribution & expectation

Recap

Last time, we have embarked on one key concept in probability: *random variables*. A random variable is simply a function that maps an outcome in a sample space into a real value. Its corresponding probability model consists of the range $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ and the probability distribution $\mathbb{P}_X(x) \;\; \forall x \in \mathcal{X}$. As a special function of random variables, we focused on *summation*, e.g., $S = X + Y$. When (X, Y) are *independent*, we showed that $\mathbb{P}_S(s)$ can be nicely represented as the *convolution* of individual probability distributions:

$$\mathbb{P}_S(s) = (\mathbb{P}_X * \mathbb{P}_Y)(s).$$

We also found that its computation can be quite complicated when (X, Y) are *dependent*. The probability model that we have studied thus far is about a *single* random variable. Of course, there is a probability model associated with *multiple* random variables.

Today's lecture

Today we will investigate such probability model. This lecture consists of five parts. In the first part, we will discuss in depth two components that form the probability model: the range and the probability distribution. It turns out the probability distribution is intimately related to the distributions of individual random variables. So such relationship would be figured out explicitly. Next we will introduce one important concept that can serve as a *representative & deterministic* quantity of a random variable. That is, *expectation*. I will explain its definition together with the rationale behind the definition. In fact, there are two key properties regarding expectation that play a powerful role in making some complicated-looking computations easy. In the 3rd part, we will investigate the first property, named the *function invariance* property. Next, we will explore the other property, called the *linearity of expectation*. Lastly we will demonstrate the power of the properties via three examples.

Probability model of multiple random variables

Consider a simple two-random-variables case: X and Y. As usual, a corresponding probability model consists of two entities. The first is the range, which in this case refers to the set of *pairs* that (X, Y) can take on. To see this clearly, first let $\mathcal{X} = \{x_1, \ldots, x_n\}$ and $\mathcal{Y} = \{y_1, \ldots, y_k\}$. Then, the range is denoted by $\mathcal{X} \times \mathcal{Y}$ and defined as:

$$\begin{aligned}\mathcal{X} \times \mathcal{Y} := \{&(x_1, y_1), (x_1, y_2), \ldots, (x_1, y_k),\\ &(x_2, y_1), (x_2, y_2), \ldots, (x_2, y_k),\\ &\;\;\vdots \qquad\quad \vdots \qquad\qquad\quad \vdots\\ &(x_n, y_1), (x_n, y_2), \ldots, (x_n, y_k)\}.\end{aligned} \tag{1}$$

Notice that its cardinality is the product of two individuals: $|\mathcal{X} \times \mathcal{Y}| = |\mathcal{X}| \cdot |\mathcal{Y}|$.

The second entity is the probability distribution, denoted by:

$$\mathbb{P}_{X,Y}(x, y) \;\; \forall (x, y) \in \mathcal{X} \times \mathcal{Y}. \tag{2}$$

Its computation is almost the same as that w.r.t. a single random variable:

$$\mathbb{P}_{X,Y}(x,y) = \mathbb{P}(X=x, Y=y) = \sum_{\omega\in\Omega:\omega\xrightarrow{X}x,\omega\xrightarrow{Y}y} \mathbb{P}(\omega) \tag{3}$$

where Ω is the sample space and the colon ":" (placed below the summation) means "such that". In order to differentiate it from the probability distribution w.r.t. a *single* random variable, people name it the *joint* probability distribution (or joint pmf). Or it is simply called the joint distribution. Since $(X,Y)=(x_1,y_1)$ and $(X,Y)=(x_2,y_2)$ are disjoint for $(x_1,y_1)\neq(x_2,y_2)$ and the union of all the events spans Ω, the sum-up-to-one constraint holds:

$$\sum_{(x,y)\in\mathcal{X}\times\mathcal{Y}} \mathbb{P}_{X,Y}(x,y) = \sum_{\omega\in\Omega}\mathbb{P}(\omega) = 1. \tag{4}$$

Relationship between $\mathbb{P}_{X,Y}(x,y)$ and $(\mathbb{P}_X(x), \mathbb{P}_Y(y))$

As mentioned earlier, there is a close relationship between $\mathbb{P}_{X,Y}(x,y)$ and $(\mathbb{P}_X(x), \mathbb{P}_Y(y))$. The relationship says:

$$\begin{aligned}\sum_{y\in\mathcal{Y}} \mathbb{P}_{X,Y}(x,y) &= \sum_{y\in\mathcal{Y}} \mathbb{P}(X=x,Y=y)\\ &= \mathbb{P}_X(x)\end{aligned}$$

where the second equality is due to the total probability law. Similarly, we get:

$$\sum_{x\in\mathcal{X}} \mathbb{P}_{X,Y}(x,y) = \mathbb{P}_Y(y).$$

This is nothing but a consequence of applying the *total probability law*. This summation process applied w.r.t. one particular random variable (marked in blue in the above) is called *marginalization*. The resulting individuals $(\mathbb{P}_X(x), \mathbb{P}_Y(y))$ are called the *marginal* distributions.

Expectation

A random variable is a value of *uncertainty*, so it can only be represented in terms of the *probability distribution* that specifies the likelihood of the occurrence of a certain value. People wanted to represent such *random* quantity in terms of a *single & deterministic* value. This is where the notion of *expectation* comes in. For a random variable X, its expectation is denoted by $\mathbb{E}[X]$ and defined as:

$$\mathbb{E}[X] := \sum_{x\in\mathcal{X}} x\cdot \mathbb{P}_X(x). \tag{5}$$

Of course, there is a reason why the expectation is defined as above. The reason is that in such a way, the expectation can be interpreted as a weighted average of all the possible values. Since the weight (marked in red in (5)) quantifies the frequency of the occurrence of x, it makes sense to view it as a *representative average*.

Two simple examples for expectation computation. First consider an experiment of rolling a dice. Let X be the dice number. Then, expectation computation is straightforward:

$$\mathbb{E}[X] = \frac{1}{6}(1+2+\cdots+6) = \frac{21}{6} = 3.5.$$

s	2	3	4	5	6	7	8	9	10	11	12
$\mathbb{P}_S(s)$	$\frac{1}{36}$	$\frac{2}{36}$	$\frac{3}{36}$	$\frac{4}{36}$	$\frac{5}{36}$	$\frac{6}{36}$	$\frac{5}{36}$	$\frac{4}{36}$	$\frac{3}{36}$	$\frac{2}{36}$	$\frac{1}{36}$

Figure 1: The probability distribution of the random variable S: the sum of the two dice number in the rolling-two-dice experiment.

Consider another experiment of rolling now two fair dice. Let S be the sum of the two dice number. Fig. 1 shows the probability distribution that we already computed in Lecture 7. This together with a tedious calculation yields:

$$\mathbb{E}[S] = \frac{1}{36}(2 \cdot 1 + 3 \cdot 2 + \cdots + 6 \cdot 5 + 7 \cdot 6 + 8 \cdot 5 + \cdots + 12 \cdot 1) = 7.$$

Property #1: Function invariance of expectation

As mentioned in the beginning, there are two important properties w.r.t. expectation. The first is the one w.r.t. a *function* of a random variable, and it is called the *function invariance* of expectation. Here is what it says. Consider a random variable, say X. A function of X, say $Y = g(X)$, satisfies:

$$\mathbb{E}[Y] = \sum_{x \in \mathcal{X}} g(x)\mathbb{P}_X(x). \tag{6}$$

Notice that the expectation of Y is computed directly via $\mathbb{P}_X(x)$ instead of $\mathbb{P}_Y(y)$. The good thing about this property is that it allows us not to rely upon the derivation $\mathbb{P}_Y(y)$ which might be cumbersome. The knowledge solely on the function $g(\cdot)$ and $\mathbb{P}_X(x)$ suffices to compute $\mathbb{E}[Y]$.

Here is the proof of (6). Starting with the definition of $\mathbb{E}[Y]$, we get:

$$\begin{aligned}
\mathbb{E}[Y] &= \sum_{y \in \mathcal{Y}} y\mathbb{P}_Y(y) \\
&= \sum_{y \in \mathcal{Y}} y\mathbb{P}(g(X) = y) \\
&= \sum_{y \in \mathcal{Y}} y \sum_{x \in \mathcal{X}: g(x) = y} \mathbb{P}_X(x)
\end{aligned} \tag{7}$$

where the 2nd equality comes from the fact that the event $Y = y$ is equivalent to $g(X) = y$; and the last equality follows the definition of the event $g(X) = y$.

Now consider the order of the summation in the last line of (7). We first fix $y \in \mathcal{Y}$ and then choose all x's such that $g(x) = y$. Next we aggregate $\mathbb{P}_X(x)$ over all such x's. We repeat this for all of the other y's in $\mathcal{Y}$. To get a concrete feel about how it is computed, let's consider one particular instance, illustrated in Fig. 2 (Left). Here (x_1, x_2) yield $g(x_1) = g(x_2) = y_1$; for others, we have $g(x_3) = g(x_4) = y_2$ and $g(x_5) = g(x_6) = y_3$. In this case, we have:

$$\begin{aligned}
&\sum_{y \in \mathcal{Y}} y \sum_{x \in \mathcal{X}: g(x) = y} \mathbb{P}_X(x) \\
&\quad = y_1(\mathbb{P}_X(x_1) + \mathbb{P}_X(x_2)) + y_2(\mathbb{P}_X(x_3) + \mathbb{P}_X(x_4)) + y_3(\mathbb{P}_X(x_5) + \mathbb{P}_X(x_6)).
\end{aligned} \tag{8}$$

Now consider the *reverse* order of the summation. We first fix $x \in \mathcal{X}$ and then find a corresponding $y = g(x)$. We then aggregate all of them over all x's. Under the same instance (see

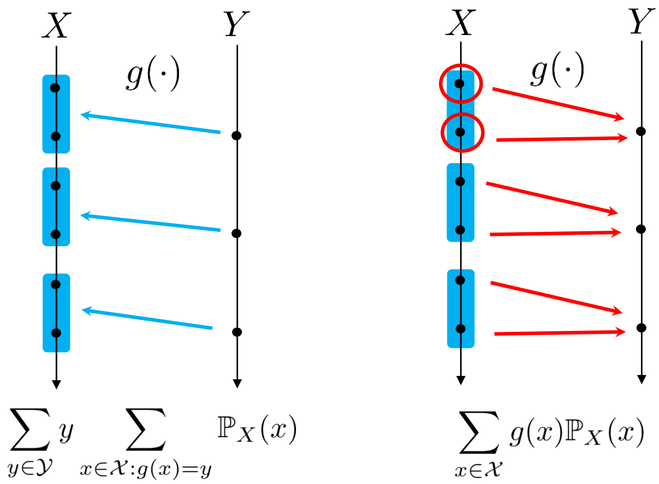

X
Y
$g(\cdot)$
$\sum_{y\in\mathcal{Y}} y \sum_{x\in\mathcal{X}:g(x)=y} \mathbb{P}_X(x)$
X
Y
$g(\cdot)$
$\sum_{x\in\mathcal{X}} g(x)\mathbb{P}_X(x)$

Figure 2: (Left): Illustration of the computation of $\sum_{y\in\mathcal{Y}} y \sum_{x\in\mathcal{X}:g(x)=y} \mathbb{P}_X(x)$. We first fix a particular value of y. Next we find all of x's such that $g(x) = y$. We then aggregate all of them over all y's; (Right): Illustration of the computation of $\sum_{x\in\mathcal{X}} g(x)\mathbb{P}_X(x)$. We first fix a particular value of x. Next we find a corresponding $y = g(x)$. We then aggregate all of them over all x's.

Fig. 2 (Right)), it can then be represented as:

$$\begin{aligned}&y_1(\mathbb{P}_X(x_1)+\mathbb{P}_X(x_2))+y_2(\mathbb{P}_X(x_3)+\mathbb{P}_X(x_4))+y_3(\mathbb{P}_X(x_5)+\mathbb{P}_X(x_6))\\&=g(x_1)\mathbb{P}_X(x_1)+g(x_2)\mathbb{P}_X(x_2)+g(x_3)\mathbb{P}_X(x_3)+g(x_4)\mathbb{P}_X(x_4)+g(x_5)\mathbb{P}_X(x_5)+g(x_6)\mathbb{P}_X(x_6).\end{aligned} \tag{9}$$

This is because $g(x_1) = g(x_2) = y_1$, $g(x_3) = g(x_4) = y_2$ and $g(x_5) = g(x_6) = y_3$. A succinct way to represent the above is:

$$\begin{aligned}&g(x_1)\mathbb{P}_X(x_1)+g(x_2)\mathbb{P}_X(x_2)+g(x_3)\mathbb{P}_X(x_3)+g(x_4)\mathbb{P}_X(x_4)+g(x_5)\mathbb{P}_X(x_5)+g(x_6)\mathbb{P}_X(x_6)\\&=\sum_{x\in\mathcal{X}} g(x)\mathbb{P}_X(x).\end{aligned} \tag{10}$$

This together with (9), (8) and (7) gives the claimed result:

$$\mathbb{E}[Y]=\sum_{y\in\mathcal{Y}} y \sum_{x\in\mathcal{X}:g(x)=y} \mathbb{P}_X(x)=\sum_{x\in\mathcal{X}} g(x)\mathbb{P}_X(x). \tag{11}$$

You may wonder whether this holds for the particular instance illustrated in Fig. 2. As you may guess, this holds in general cases as well. The reason is similar to the one that I offered in Lecture 7 in the process of explaining two observations w.r.t. the sum-up-to-one constraint of a random variable. The reason is that (i) all possible y's span the entire domain $\mathcal{X}$, i.e., for every $x \in \mathcal{X}$, there always exists a particular y; and (ii) one-to-many mapping of a function is invalid, i.e., each $x \in \mathcal{X}$ yields the *unique* output y.

Property #2: Linearity of expectation

The second is a very powerful property, named the *linearity* of expectation. The property consists of two subproperties. The first is the *additivity* property:

$$\mathbb{E}[X+Y]=\mathbb{E}[X]+\mathbb{E}[Y]. \tag{12}$$

Here the expectation in the left-hand-side is w.r.t. $\mathbb{P}_{X,Y}(x,y)$. On the other hand, the expectations in the right-hand-side are w.r.t. $\mathbb{P}_X(x)$ and $\mathbb{P}_Y(y)$, respectively. The second is the *homogeneity* property:

$$\mathbb{E}[cX] = c\mathbb{E}[X] \text{ for any constant } c. \tag{13}$$

The proofs are straightforward. First we obtain:

$$\begin{aligned}
\mathbb{E}[X+Y] &= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}}(x+y)\mathbb{P}_{X,Y}(x,y) \\
&= \sum_{x\in\mathcal{X}} x \sum_{y\in\mathcal{Y}} \mathbb{P}_{X,Y}(x,y) + \sum_{y\in\mathcal{Y}} y \sum_{x\in\mathcal{Y}} \mathbb{P}_{X,Y}(x,y) \\
&= \sum_{x\in\mathcal{X}} x\mathbb{P}_X(x) + \sum_{y\in\mathcal{Y}} y\mathbb{P}_Y(y) \\
&= \mathbb{E}[X] + \mathbb{E}[Y]
\end{aligned}$$

where the 1st and last equalities are due to the definition of expectation; and the 3rd follows from the total probability law. For the proof of homogeneity, we get:

$$\begin{aligned}
\mathbb{E}[cX] &= \sum_{x\in\mathcal{X}}(cx)\mathbb{P}_X(x) \\
&= c\sum_{x\in\mathcal{X}} x\mathbb{P}_X(x) = c\mathbb{E}[X]
\end{aligned}$$

where the 1st equality is due to the *function invariance property* (6).

It turns out the linearity property (reflected in (12) and (13)) plays a powerful role in making many complicated-looking computations tractable. Here we focus on three such examples.

Example #1: Rolling two dice

The first example is the one that we investigated earlier in this lecture: Rolling two fair dice. Consider $S = X + Y$ where X and Y indicate the 1st and 2nd dice number, respectively. Previously, we computed $\mathbb{E}[S]$ via $\mathbb{P}_S(s)$. However, by exploiting the linearity property particularly (12), we can simplify the computation as:

$$\mathbb{E}[S] = \mathbb{E}[X] + \mathbb{E}[Y] = 3.5 + 3.5 = 7.$$

Example #2: Tossing a p-biased coin n times

The second is the example that we explored in the previous lecture: Tossing a p-biased coin n times. Consider the total number of "Head"s: S. Focus on $\mathbb{E}[S]$. Remember that we expressed S as the sum of the following basic random variables:

$$\begin{aligned}
&X_i = \begin{cases} 1, & \text{if the } i\text{th flips shows "Head";} \\ 0, & \text{otherwise;} \end{cases} \\
&S = X_1 + X_2 + \cdots + X_n.
\end{aligned}$$

We also computed the probability distribution of S:

$$\mathbb{P}_S(s) = \binom{n}{s} p^s (1-p)^{n-s}, \qquad \forall s \in \mathcal{S} = \{0, 1, \ldots, n\}.$$

One naive way to compute $\mathbb{E}[S]$ is to simply apply the definition of expectation with $\mathbb{P}_S(s)$:

$$\mathbb{E}[S] = \sum_{s=0}^{n} s \cdot \binom{n}{s} p^s (1-p)^{n-s}. \tag{14}$$

How do you feel about (14)? Do you want to try it? If I were you, I would not. Instead, exploiting again the linearity of expectation, one can greatly simplify the computation as:

$$\mathbb{E}[S] = \mathbb{E}[X_1] + \cdots + \mathbb{E}[X_n] = np \tag{15}$$

where the 2nd equality follows from $\mathbb{E}[X_i] = p \cdot 1 + (1-p) \cdot 0 = p$, $\forall i \in \{1, 2, \dots, n\}$.

Example #3: Homework matching

The last example is the one in which we gave up computing the probability distribution: the *homework matching problem.* See Fig. 3. Consider the number of students S who receive their own homeworks. Remember we expressed S as the sum of the following basic random variables:

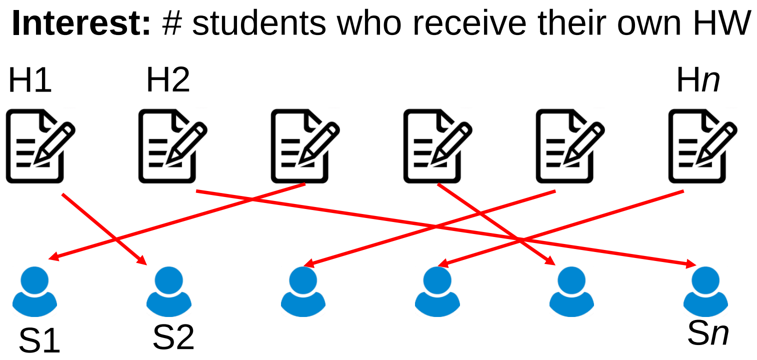


Figure 3: Homework matching problem: The homeworks collected from n students are shuffled and returned back to the students. Each student receives only one homework (one-to-one matching), yet the returned homework is not necessarily his/her own homework.

$$X_i = \begin{cases} 1, & \text{if the } i\text{th student receives her own homework;} \\ 0, & \text{otherwise;} \end{cases}$$
$$S = X_1 + X_2 + \cdots + X_n.$$

In the previous lecture, I mentioned that the dependence of X_i's makes the computation of $\mathbb{P}_S(s)$ intractable. That's why we gave up computing $\mathbb{P}_S(s)$. Here the linearity of expectation comes to rescue:

$$\mathbb{E}[S] = \mathbb{E}[X_1] + \cdots + \mathbb{E}[X_n] = \frac{1}{n} \cdot n = 1$$

where the 2nd equality follows from $\mathbb{E}[X_i] = \frac{1}{n}$ $\forall i \in \{1, 2, \dots, n\}$.

Look ahead

In this lecture, I claimed that expectation serves as a deterministic quantity that can *well* represent a random variable with uncertainty. But the expectation might not be quite representative especially when the interested random variable takes many different values with high chances.

One such extreme example might be the number X chosen from $\{1, 2, \ldots, 100\}$ uniformly at random. In this case, $\mathbb{E}[X] = \frac{1+2\cdots+100}{100} = 50.5$ might be far away from one particular realization, say 13, which has the same chance among 100 possibilities. Then, can we say $\mathbb{E}[X]$ *well represent* a random number X? It turns out there is a measure that quantifies the degree of well representing. Next time, we will investigate the measure: *variance*.

EE210 Probability and Introductory Random Processes — March 30, 2021
KAIST, Spring 2021 — Changho Suh (chsuh@kaist.ac.kr)

Lecture 9: Variance

Recap

Last time, we studied the notion of expectation that can serve as a deterministic representative for a random variable with uncertainty. Its definition reads:

$$\mathbb{E}[X] := \sum_{x \in \mathcal{X}} x \cdot \mathbb{P}_X(x). \tag{1}$$

We also investigated two important properties. The first is the *function invariance* property that plays a role in enabling the direct computation of $\mathbb{E}[Y]$ without relying upon the computation of $\mathbb{P}_Y(y)$ for $Y = g(X)$. The property says:

$$\mathbb{E}[Y] = \sum_{x \in \mathcal{X}} g(x)\mathbb{P}_X(x). \tag{2}$$

The second is the *linearity* property consisting of two subproperties:

$$\begin{aligned} &\text{(Additivity): } \mathbb{E}[X+Y] = \mathbb{E}[X] + \mathbb{E}[Y]; \\ &\text{(Homogeneity): } \mathbb{E}[cX] = c\mathbb{E}[X] \;\text{ for any constant } c. \end{aligned} \tag{3}$$

At the end of the last lecture, I raised one natural question: Can the expectation *well represent* the uncertain random variable? I then claimed that there is a measure that quantifies the degree of well representing. The measure is *variance*.

Today's lecture

Today we will explore details on the variance. This lecture consists of four parts. In the first part, we will introduce the definition of variance together with the rationale behind the definition. It turns out there is a very useful fact which allows us to compute the variance efficiently. So in the second part, we will study the useful fact. Next we will study two important properties that play significant roles in addressing some challenging situations, wherein the variance computation is very difficult if we rely solely upon the definition. Lastly we will explore one prominent inequality that serves to quantify the degree of well representing more precisely, relative to the variance itself. That is, *Chebyshev's inequality*.

Definition of variance

Let's start from the beginning. For a random variable X, its variance is denoted by $\mathsf{Var}(X)$ and defined as:

$$\mathsf{Var}(X) := \mathbb{E}[(X-\mu)^2] \tag{4}$$

where $\mu := \mathbb{E}[X]$. As usual, there must be a reason that the variance is defined as above. At first glance, this definition is a bit unnatural. A more natural definition might be $\mathbb{E}[X-\mu]$, as it indicates indeed the *averaged deviation* from the center μ. But this is not informative at all, as $\mathbb{E}[X-\mu]$ is always 0 (due to the linearity of expectation (3)) no matter what $\mathbb{P}_X(x)$ is. The reason of $\mathbb{E}[X-\mu]$ being 0 all the time is that the sign of $X-\mu$ is either plus or minus with the same frequency. So as another natural definition, people thought of a measure that takes always a plus sign. The measure was $\mathbb{E}[|X-\mu|]$. Now it is never 0 unless X is a deterministic constant. So it can act as a proper measure. However, this was not selected because people found its computation often very difficult. As the 3rd candidate, people came up with $\mathsf{Var}(X) := \mathbb{E}[(X-\mu)^2]$ and tried many instances to check whether its computation is more-or-less okay. It turns out the computation is often tractable. This is the sole reason why the variance is defined as (4). I am sure that the choice was $\mathbb{E}[|X-\mu|]$ if its computation were not that difficult.

However, there was still an issue on the definition (4). Due to the power 2 in the definition, the *unit* of $\mathsf{Var}(X)$ does not match with that of X. If the unit of X is meter, then the unit of $\mathsf{Var}(X)$ would be meter^2. So it does not properly capture the *deviation* from the center. This is where another measure that you may hear of comes in. That is, *standard deviation*, simply defined as the square root of variance:

$$\sigma(X) := \sqrt{\mathsf{Var}(X)}. \tag{5}$$

Actually, this is a bit annoying situation. Sometimes, mathematicians introduce many seemingly-redundant notions (like variance and standard deviation) only because of the *beauty of math*. They pursue the beauty and tractability, and variance is the one that comes as a consequence of such an attitude.

A useful fact

There is a very useful fact which allows us to compute the variance a bit more efficiently. The fact is:

$$\mathsf{Var}(X) = \mathbb{E}[X^2] - \mu^2. \tag{6}$$

The proof of this is straightforward. Starting with the definition of variance, we get:

$$\begin{aligned}\mathsf{Var}(X) &:= \mathbb{E}[(X-\mu)^2] \\ &= \mathbb{E}[X^2 - 2\mu X + \mu^2] \\ &= \mathbb{E}[X^2] + \mathbb{E}[-2\mu X] + \mathbb{E}[\mu^2] \\ &= \mathbb{E}[X^2] - 2\mu\mathbb{E}[X] + \mathbb{E}[\mu^2] \\ &= \mathbb{E}[X^2] - \mu^2\end{aligned}$$

where the 3rd and 4th steps are due to the additivity and homogeneity properties of expectation (3), respectively. It turns out in many cases, the computation of $\mathbb{E}[X^2]$ is a bit easier than that of its translated version $\mathbb{E}[(X-\mu)^2]$. That's why people often employ (6) instead of the original definition (4).

Here is one such example where the computation of $\mathbb{E}[X^2]$ is indeed easier. Let X be a uniformly distributed random variable that takes one value from $\mathcal{X} := \{1, 2, \ldots, n\}$. Let $\mathbb{P}_X(x) = \frac{1}{n}$ be the corresponding probability distribution where $x \in \mathcal{X}$. The expectation reads $\mathbb{E}[X] = \frac{1}{n}(1+2+\cdots+n) = \frac{n+1}{2}$. The expectation of X^2 (or called the 2nd moment) is:

$$\begin{aligned}\mathbb{E}[X^2] &= \frac{1}{n}(1^2+2^2+\cdots+n^2) \\ &= \frac{1}{n}\cdot\frac{n(n+1)(2n+1)}{6} \\ &= \frac{(n+1)(2n+1)}{6}\end{aligned}$$

where the 2nd equality follows from the fact that you learned from calculus: $\sum_{i=1}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}$ (Did you forget how to prove this? If so, think about $f(n+1) - f(n)$ where $f(n) := n^3$). Hence,

$$\begin{aligned}\mathsf{Var}(X) &= \mathbb{E}[X^2] - (\mathbb{E}[X])^2 \\ &= \frac{(n+1)(2n+1)}{6} - \frac{(n+1)^2}{4} \\ &= (n+1)\left(\frac{n-1}{12}\right) = \frac{n^2-1}{12}.\end{aligned}$$

Two important properties

As mentioned in the beginning, there are two important properties that help us to address challenging situations where a direct way of computing variance is non-trivial. These are about *independent* random variables. For illustrative purpose, let's focus on a simple setting where there are two independent random variables, say X and Y. The first property is the one called the *uncorrelatedness*, which says that the expectation of the product is the product of individuals:

$$\mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y]. \tag{7}$$

The second is the one called the *additivity*, which says that the variance of the sum is the sum of individuals:

$$\mathsf{Var}(X+Y) = \mathsf{Var}(X) + \mathsf{Var}(Y). \tag{8}$$

The proofs of these are not that difficult. First let's prove (7). Starting with the definition of expectation, we get:

$$\begin{aligned}\mathbb{E}[XY] &= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} xy\mathbb{P}_{X,Y}(x,y) \\ &= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} xy\mathbb{P}_X(x)\mathbb{P}_Y(y) \\ &= \sum_{x\in\mathcal{X}} x\mathbb{P}_X(x)\sum_{y\in\mathcal{Y}} y\mathbb{P}_Y(y) \\ &= \mathbb{E}[X]\mathbb{E}[Y]\end{aligned}$$

where the 2nd equality is due to the independence of (X,Y). The proof of (8) is also easy particularly with the help of (7). Let $\mu_X = \mathbb{E}[X]$ and $\mu_Y = \mathbb{E}[Y]$. Starting with the useful fact (6), we have:

$$\begin{aligned}\mathsf{Var}(X+Y) &= \mathbb{E}[(X+Y)^2] - (\mu_X+\mu_Y)^2 \\ &= \mathbb{E}[X^2] + \mathbb{E}[Y^2] + 2\mathbb{E}[XY] - (\mu_X+\mu_Y)^2 \\ &= \mathbb{E}[X^2] + \mathbb{E}[Y^2] + 2\mathbb{E}[X]\mathbb{E}[Y] - (\mu_X+\mu_Y)^2 \\ &= \mathbb{E}[X^2] - \mu_X^2 + \mathbb{E}[Y^2] - \mu_Y^2 \\ &= \mathsf{Var}(X) + \mathsf{Var}(Y)\end{aligned}$$

where the 2nd equality is due to the linearity of expectation (3); the 3rd equality follows from the uncorrelateness property (7).

Example: Tossing a p-biased coin n times

Here is one example wherein the two properties (the second property in particular) play a role. The example is the one that we frequently explored in the past lectures: Tossing a p-biased coin n times. Let S be the total number of "Head"s. Here the focus is to compute $\mathsf{Var}(S)$. Using the useful fact (6), we obtain:

$$\mathsf{Var}(S) = \mathbb{E}[S^2] - (\mathbb{E}[S])^2. \tag{9}$$

Since we have already figured out from Lecture 7 that S follows the prominent Binomial distribution, one natural way is to compute $(\mathbb{E}[S], \mathbb{E}[S^2])$ (called the 1st and 2nd moments) and then to plug them into the above (9). But this way turns out to be by far much complicated, relative to the one that we will take in the sequel.

The different way that we will take is to exploit the additivity property (8). To this end, as we did before, we first express S as the sum of the following basic random variables:

$$S = X_1 + X_2 + \cdots + X_n$$

where $X_i = \mathbf{1}\{i\text{th flips shows "Head"}\}$. Here the key observation is that X_i's are *independent*. Hence, by applying (8) several times ($n-1$ times precisely), we get:

$$\begin{aligned}\mathsf{Var}(S) &= \mathsf{Var}(X_1) + \mathsf{Var}(X_2 + \cdots + X_n)\\ &= \mathsf{Var}(X_1) + \mathsf{Var}(X_2) + \mathsf{Var}(X_3 + \cdots + X_n)\\ &\ \ \vdots\\ &= \mathsf{Var}(X_1) + \mathsf{Var}(X_2) + \cdots + \mathsf{Var}(X_n).\end{aligned} \tag{10}$$

Here, the variance of one particular random variable, say X_1, is:

$$\mathsf{Var}(X_1) = \mathbb{E}[X_1^2] - (\mathbb{E}[X_1])^2 = p - p^2 = p(1-p). \tag{11}$$

By symmetry, $\mathsf{Var}(X_1) = \cdots = \mathsf{Var}(X_n)$. This together with (10) yields:

$$\mathsf{Var}(S) = \mathsf{Var}(X_1) + \cdots + \mathsf{Var}(X_n) = np(1-p).$$

A more precise measure of how well $\mathbb{E}[X]$ represents X

Remember that we introduced the variance, (more properly saying, the standard deviation for the purpose of matching the unit), as a measure that quantifies how well the expectation $\mathbb{E}[X]$ represents the uncertain random variable X:

$$\sigma(X) := \sqrt{\mathsf{Var}(X)}, \quad \mathsf{Var}(X) := \mathbb{E}[(X-\mu)^2].$$

But there is a limitation on this measure. The measure quantifies only the degree of the spreadness around the center. It does not translate to a concrete *probabilistic* number. For instance, someone may be interested in the explicit probability numbers like:

The probability of how often X deviates from μ by a certain distance, say d.

The above probability, simply called the *tail probability*, can be formally written as:

$$\mathbb{P}(|X-\mu| \geq d). \tag{12}$$

The smaller this measure is, the higher representation capability. Hence, the tail probability can serve as a more precise measure. Now how to compute (12)? It turns out the computation of (12) is often very difficult. What can we do then? As I mentioned in Lecture 6, whenever facing with challenges in the probability world, the idea for overcoming the challenges is to *relax the goal.* Here what it means by relaxing the goal is to obtain a bound on the tail probability. Which bound is helpful between upper and lower bounds? Here having a smaller tail probability, the better situation we have in terms of representation capability. Hence, an *upper* bound helps because a smaller upper bound shrinks down the tail probability. There is a prominent inequality that offers an upper bound on the tail probability. That is, *Chebyshev's inequality*:

$$\mathbb{P}(|X-\mu| \geq d) \leq \frac{\mathsf{Var}(X)}{d^2}. \tag{13}$$

This inequality makes an intuitive sense. A small variance yields a small upper bound, forcing the tail probability to decrease. A large deviation d from the center should give obviously a small chance of $|X-\mu| \geq d$. This is well reflected in the inequality.

Proof of Chebyshev's inequality (13)

There are different proofs for Chebyshev's inequality (13). Here I will provide one elementary-level proof which requires only the definitions while not relying upon any other new techniques. In PS3, you will have a chance to try another proof that requires another bounding technique. Although the proof considered herein hinges solely upon the definitions, it contains a certain non-trivial trick. The trick is to smartly define a new random variable, say Y, from X. We define Y such that $\mathbb{P}_Y(y)$ is the same as $\mathbb{P}_X(y)$ when $|y-\mu| < d$; takes an aggregated probability $\mathbb{P}(X-\mu \geq d)$ when $y-\mu = d$; and takes the other-side aggregated probability $\mathbb{P}(X-\mu \leq -d)$ when $y-\mu=-d$:

$$\mathbb{P}_Y(y) = \begin{cases} \mathbb{P}_X(y), & \text{if } |y-\mu| < d; \\ \mathbb{P}(X-\mu \geq d), & \text{if } y-\mu = d; \\ \mathbb{P}(X-\mu \leq -d), & \text{if } y-\mu=-d; \\ 0, & \text{if } |y-\mu| > d. \end{cases} \tag{14}$$

See Fig. 1 for a clearer illustration. You will soon figure out why such Y helps in the proof. First observe that

$$\mathsf{Var}(X) \geq \mathsf{Var}(Y). \tag{15}$$

This is immediate because Y is more densely located around μ. If you are not convinced about this sort of intuitive reasoning, you may want to do a rigorous proof via massaging $\mathsf{Var}(X)$ to relate to $\mathsf{Var}(Y)$ with the help of (14). Please try it if you feel so. With (15) and (13), it suffices to show that

$$\mathsf{Var}(Y) \geq d^2\mathbb{P}(|X-\mu| \geq d). \tag{16}$$

This can be proved via the following steps:

$$\begin{aligned} \mathsf{Var}(Y) &= \sum_{y:|y-\mu|<d} (y-\mu)^2\mathbb{P}_Y(y) + \sum_{y:|y-\mu|\geq d} (y-\mu)^2\mathbb{P}_Y(y) \\ &\geq \sum_{y:|y-\mu|\geq d} (y-\mu)^2\mathbb{P}_Y(y) \\ &\geq d^2 \sum_{y:|y-\mu|\geq d} \mathbb{P}_Y(y) \\ &= d^2\mathbb{P}(|X-\mu| \geq d) \end{aligned}$$

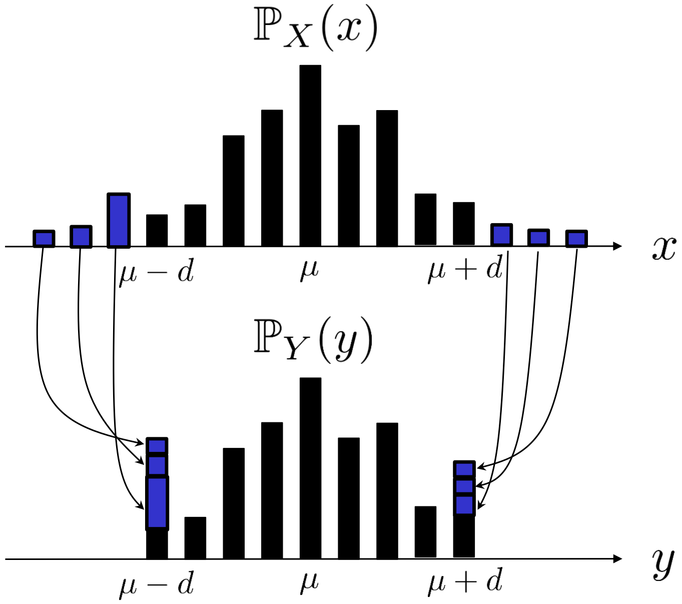


Figure 1: (Top): Probability distribution of a random variable X; (Bottom): Probability distribution of a newly defined random variable Y. We define Y such that $\mathbb{P}_Y(y)$ is the same as $\mathbb{P}_X(y)$ when $|y-\mu| \le d$; takes an aggregated probability $\mathbb{P}(X-\mu \ge d)$ when $y-\mu = d$; and takes the other-side aggregated probability $\mathbb{P}(X-\mu \le -d)$ when $y-\mu = -d$.

where the 2nd last step is because $(y-\mu)^2 \ge d^2$ in the considered range of summation; and the last equality follows from $\mathbb{P}_Y(\mu+d) = \mathbb{P}(X-\mu \ge d)$ and $\mathbb{P}_Y(\mu-d) = \mathbb{P}(X-\mu \le -d)$ due to (14).

Look ahead

So far we have considered a particular type of random variables: *discrete* random variables. In reality, however, there are many scenarios concerning *continuous-valued* random quantities. One such scenario is *communication* that we will touch upon in great details in Part III: Applications. As I mentioned in Lecture 1, there is an enemy in communication systems. The enemy is *noise* which can be modeled as a continuous-valued random signal. There is a relevant concept concerning such continuous random signal. That is, *continuous* random variable. So next time, we will investigate the continuous random variable.

EE210 Probability and Introductory Random Processes
KAIST, Spring 2021

March 30, 2021
Changho Suh (chsuh@kaist.ac.kr)

Lecture 9: Variance

Recap

Last time, we studied the notion of expectation that can serve as a deterministic representative for a random variable with uncertainty. Its definition reads:

$$\mathbb{E}[X] := \sum_{x \in \mathcal{X}} x \cdot \mathbb{P}_X(x). \tag{1}$$

We also investigated two important properties. The first is the *function invariance* property that plays a role in enabling the direct computation of $\mathbb{E}[Y]$ without relying upon the computation of $\mathbb{P}_Y(y)$ for $Y = g(X)$. The property says:

$$\mathbb{E}[Y] = \sum_{x \in \mathcal{X}} g(x) \mathbb{P}_X(x). \tag{2}$$

The second is the *linearity* property consisting of two subproperties:

$$\begin{aligned} &\text{(Additivity): } \mathbb{E}[X+Y] = \mathbb{E}[X] + \mathbb{E}[Y]; \\ &\text{(Homogeneity): } \mathbb{E}[cX] = c\mathbb{E}[X] \text{ for any constant } c. \end{aligned} \tag{3}$$

At the end of the last lecture, I raised one natural question: Can the expectation *well represent* the uncertain random variable? I then claimed that there is a measure that quantifies the degree of well representing. The measure is *variance.*

Today's lecture

Today we will explore details on the variance. This lecture consists of four parts. In the first part, we will introduce the definition of variance together with the rationale behind the definition. It turns out there is a very useful fact which allows us to compute the variance efficiently. So in the second part, we will study the useful fact. Next we will study two important properties that play significant roles in addressing some challenging situations, wherein the variance computation is very difficult if we rely solely upon the definition. Lastly we will explore one prominent inequality that serves to quantify the degree of well representing more precisely, relative to the variance itself. That is, *Chebyshev's inequality.*

Definition of variance

Let's start from the beginning. For a random variable X, its variance is denoted by $\mathsf{Var}(X)$ and defined as:

$$\mathsf{Var}(X) := \mathbb{E}[(X-\mu)^2] \tag{4}$$

where $\mu := \mathbb{E}[X]$. As usual, there must be a reason that the variance is defined as above. At first glance, this definition is a bit unnatural. A more natural definition might be $\mathbb{E}[X - \mu]$, as it indicates indeed the *averaged deviation* from the center μ. But this is not informative at all, as $\mathbb{E}[X - \mu]$ is always 0 (due to the linearity of expectation (3)) no matter what $\mathbb{P}_X(x)$ is. The reason of $\mathbb{E}[X - \mu]$ being 0 all the time is that the sign of $X - \mu$ is either plus or minus with the same frequency. So as another natural definition, people thought of a measure that takes always a plus sign. The measure was $\mathbb{E}[|X-\mu|]$. Now it is never 0 unless X is a deterministic constant. So it can act as a proper measure. However, this was not selected because people found its computation often very difficult. As the 3rd candidate, people came up with $\mathsf{Var}(X) := \mathbb{E}[(X-\mu)^2]$ and tried many instances to check whether its computation is more-or-less okay. It turns out the computation is often tractable. This is the sole reason why the variance is defined as (4). I am sure that the choice was $\mathbb{E}[|X-\mu|]$ if its computation were not that difficult.

However, there was still an issue on the definition (4). Due to the power 2 in the definition, the *unit* of $\mathsf{Var}(X)$ does not match with that of X. If the unit of X is meter, then the unit of $\mathsf{Var}(X)$ would be meter^2. So it does not properly capture the *deviation* from the center. This is where another measure that you may hear of comes in. That is, *standard deviation*, simply defined as the square root of variance:

$$\sigma(X) := \sqrt{\mathsf{Var}(X)}. \tag{5}$$

Actually, this is a bit annoying situation. Sometimes, mathematicians introduce many seemingly-redundant notions (like variance and standard deviation) only because of the *beauty of math*. They pursue the beauty and tractability, and variance is the one that comes as a consequence of such an attitude.

A useful fact

There is a very useful fact which allows us to compute the variance a bit more efficiently. The fact is:

$$\mathsf{Var}(X) = \mathbb{E}[X^2] - \mu^2. \tag{6}$$

The proof of this is straightforward. Starting with the definition of variance, we get:

$$\begin{aligned}\mathsf{Var}(X) &:= \mathbb{E}[(X-\mu)^2]\\ &= \mathbb{E}[X^2 - 2\mu X + \mu^2]\\ &= \mathbb{E}[X^2] + \mathbb{E}[-2\mu X] + \mathbb{E}[\mu^2]\\ &= \mathbb{E}[X^2] - 2\mu\mathbb{E}[X] + \mathbb{E}[\mu^2]\\ &= \mathbb{E}[X^2] - \mu^2\end{aligned}$$

where the 3rd and 4th steps are due to the additivity and homogeneity properties of expectation (3), respectively. It turns out in many cases, the computation of $\mathbb{E}[X^2]$ is a bit easier than that of its translated version $\mathbb{E}[(X-\mu)^2]$. That's why people often employ (6) instead of the original definition (4).

Here is one such example where the computation of $\mathbb{E}[X^2]$ is indeed easier. Let X be a uniformly distributed random variable that takes one value from $\mathcal{X} := \{1, 2, \ldots, n\}$. Let $\mathbb{P}_X(x) = \frac{1}{n}$ be the corresponding probability distribution where $x \in \mathcal{X}$. The expectation reads $\mathbb{E}[X] = \frac{1}{n}(1 + 2 + \cdots + n) = \frac{n+1}{2}$. The expectation of X^2 (or called the 2nd moment) is:

$$\begin{aligned}\mathbb{E}[X^2] &= \frac{1}{n}(1^2 + 2^2 + \cdots + n^2)\\ &= \frac{1}{n}\cdot\frac{n(n+1)(2n+1)}{6}\\ &= \frac{(n+1)(2n+1)}{6}\end{aligned}$$

where the 2nd equality follows from the fact that you learned from calculus: $\sum_{i=1}^{n} i^2 = \frac{n(n+1)(2n+1)}{6}$ (Did you forget how to prove this? If so, think about $f(n+1) - f(n)$ where $f(n) := n^3$). Hence,

$$\begin{aligned}\mathsf{Var}(X) &= \mathbb{E}[X^2] - (\mathbb{E}[X])^2 \\ &= \frac{(n+1)(2n+1)}{6} - \frac{(n+1)^2}{4} \\ &= (n+1)\left(\frac{n-1}{12}\right) = \frac{n^2-1}{12}.\end{aligned}$$

Two important properties

As mentioned in the beginning, there are two important properties that help us to address challenging situations where a direct way of computing variance is non-trivial. These are about *independent* random variables. For illustrative purpose, let's focus on a simple setting where there are two independent random variables, say X and Y. The first property is the one called the *uncorrelatedness*, which says that the expectation of the product is the product of individuals:

$$\mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y]. \tag{7}$$

The second is the one called the *additivity*, which says that the variance of the sum is the sum of individuals:

$$\mathsf{Var}(X+Y) = \mathsf{Var}(X) + \mathsf{Var}(Y). \tag{8}$$

The proofs of these are not that difficult. First let's prove (7). Starting with the definition of expectation, we get:

$$\begin{aligned}\mathbb{E}[XY] &= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} xy\mathbb{P}_{X,Y}(x,y) \\ &= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} xy\mathbb{P}_X(x)\mathbb{P}_Y(y) \\ &= \sum_{x\in\mathcal{X}} x\mathbb{P}_X(x)\sum_{y\in\mathcal{Y}} y\mathbb{P}_Y(y) \\ &= \mathbb{E}[X]\mathbb{E}[Y]\end{aligned}$$

where the 2nd equality is due to the independence of (X,Y). The proof of (8) is also easy particularly with the help of (7). Let $\mu_X = \mathbb{E}[X]$ and $\mu_Y = \mathbb{E}[Y]$. Starting with the useful fact (6), we have:

$$\begin{aligned}\mathsf{Var}(X+Y) &= \mathbb{E}[(X+Y)^2] - (\mu_X+\mu_Y)^2 \\ &= \mathbb{E}[X^2] + \mathbb{E}[Y^2] + 2\mathbb{E}[XY] - (\mu_X+\mu_Y)^2 \\ &= \mathbb{E}[X^2] + \mathbb{E}[Y^2] + 2\mathbb{E}[X]\mathbb{E}[Y] - (\mu_X+\mu_Y)^2 \\ &= \mathbb{E}[X^2] - \mu_X^2 + \mathbb{E}[Y^2] - \mu_Y^2 \\ &= \mathsf{Var}(X) + \mathsf{Var}(Y)\end{aligned}$$

where the 2nd equality is due to the linearity of expectation (3); the 3rd equality follows from the uncorrelateness property (7).

Example: Tossing a p-biased coin n times

Here is one example wherein the two properties (the second property in particular) play a role. The example is the one that we frequently explored in the past lectures: Tossing a p-biased coin n times. Let S be the total number of "Head"s. Here the focus is to compute $\mathsf{Var}(S)$. Using the useful fact (6), we obtain:

$$\mathsf{Var}(S) = \mathbb{E}[S^2] - (\mathbb{E}[S])^2. \tag{9}$$

Since we have already figured out from Lecture 7 that S follows the prominent Binomial distribution, one natural way is to compute $(\mathbb{E}[S], \mathbb{E}[S^2])$ (called the 1st and 2nd moments) and then to plug them into the above (9). But this way turns out to be by far much complicated, relative to the one that we will take in the sequel.

The different way that we will take is to exploit the additivity property (8). To this end, as we did before, we first express S as the sum of the following basic random variables:

$$S = X_1 + X_2 + \cdots + X_n$$

where $X_i = \mathbf{1}\{i\text{th flips shows "Head"}\}$. Here the key observation is that X_i's are *independent*. Hence, by applying (8) several times ($n-1$ times precisely), we get:

$$\begin{aligned}\mathsf{Var}(S) &= \mathsf{Var}(X_1) + \mathsf{Var}(X_2 + \cdots + X_n)\\ &= \mathsf{Var}(X_1) + \mathsf{Var}(X_2) + \mathsf{Var}(X_3 + \cdots + X_n)\\ &\;\;\vdots\\ &= \mathsf{Var}(X_1) + \mathsf{Var}(X_2) + \cdots + \mathsf{Var}(X_n).\end{aligned} \tag{10}$$

Here, the variance of one particular random variable, say X_1, is:

$$\mathsf{Var}(X_1) = \mathbb{E}[X_1^2] - (\mathbb{E}[X_1])^2 = p - p^2 = p(1-p). \tag{11}$$

By symmetry, $\mathsf{Var}(X_1) = \cdots = \mathsf{Var}(X_n)$. This together with (10) yields:

$$\mathsf{Var}(S) = \mathsf{Var}(X_1) + \cdots + \mathsf{Var}(X_n) = np(1-p).$$

A more precise measure of how well $\mathbb{E}[X]$ represents X

Remember that we introduced the variance, (more properly saying, the standard deviation for the purpose of matching the unit), as a measure that quantifies how well the expectation $\mathbb{E}[X]$ represents the uncertain random variable X:

$$\sigma(X) := \sqrt{\mathsf{Var}(X)}, \quad \mathsf{Var}(X) := \mathbb{E}[(X-\mu)^2].$$

But there is a limitation on this measure. The measure quantifies only the degree of the spreadness around the center. It does not translate to a concrete *probabilistic* number. For instance, someone may be interested in the explicit probability numbers like:

The probability of how often X deviates from μ by a certain distance, say d.

The above probability, simply called the *tail probability*, can be formally written as:

$$\mathbb{P}(|X-\mu| \geq d). \tag{12}$$

The smaller this measure is, the higher representation capability. Hence, the tail probability can serve as a more precise measure. Now how to compute (12)? It turns out the computation of (12) is often very difficult. What can we do then? As I mentioned in Lecture 6, whenever facing with challenges in the probability world, the idea for overcoming the challenges is to *relax the goal.* Here what it means by relaxing the goal is to obtain a bound on the tail probability. Which bound is helpful between upper and lower bounds? Here having a smaller tail probability, the better situation we have in terms of representation capability. Hence, an *upper* bound helps because a smaller upper bound shrinks down the tail probability. There is a prominent inequality that offers an upper bound on the tail probability. That is, *Chebyshev's inequality*:

$$\mathbb{P}(|X-\mu| \geq d) \leq \frac{\mathsf{Var}(X)}{d^2}. \tag{13}$$

This inequality makes an intuitive sense. A small variance yields a small upper bound, forcing the tail probability to decrease. A large deviation d from the center should give obviously a small chance of $|X-\mu| \geq d$. This is well reflected in the inequality.

Proof of Chebyshev's inequality (13)

The proof is very easy once we employ another popular inequality, named *Markov' inequality*: for a nonnegative random variable Y and $d > 0$,

$$\mathbb{P}(Y \geq d) \leq \frac{\mathbb{E}[Y]}{d}. \tag{14}$$

Using Markov's inequality, we get:

$$\begin{aligned}\mathbb{P}(|X-\mu| \geq d) &= \mathbb{P}((X-\mu)^2 \geq d^2)\\ &\leq \frac{\mathbb{E}[(X-\mu)^2]}{d^2} = \frac{\mathsf{Var}(X)}{d^2}\end{aligned}$$

where the 1st equality follows from the fact that $\{(X-\mu)^2 \geq d^2\}$ is an equivalent event to $\{|X-\mu| \geq d\}$; and the last step is due to the definition of variance.

Proof of Markov's inequality (14)

We start with a key observation:

$$Y \geq d \cdot \mathbf{1}\{Y \geq d\}. \tag{15}$$

This is immediate because the RHS reads 0 when $Y < d$ (while the LHS is $Y \geq 0$) ; otherwise, i.e., when $Y \geq d$, the RHS reads d while the LHS is $Y \geq d$. Since (15) holds for any Y, the inequality still holds when taking expectation on both sides:

$$\begin{aligned}\mathbb{E}[Y] &\geq d \cdot \mathbb{E}[\mathbf{1}\{Y \geq d\}]\\ &= d \cdot \mathbb{P}(Y \geq d).\end{aligned}$$

Hence, we complete the proof.

Look ahead

So far we have considered a particular type of random variables: *discrete* random variables. In reality, however, there are many scenarios concerning *continuous-valued* random quantities. One such scenario is *communication* that we will touch upon in great details in Part III: Applications. As I mentioned in Lecture 1, there is an enemy in communication systems. The enemy is *noise* which can be modeled as a continuous-valued random signal. There is a relevant concept concerning such continuous random signal. That is, *continuous* random variable. So next time, we will investigate the continuous random variable.

EE210 Probability and Introductory Random Processes April 1, 2021
KAIST, Spring 2021 Changho Suh (chsuh@kaist.ac.kr)

Lecture 10: Continuous random variables

Recap

Last time, we learned about another concept in probability: *variance*, which is shown to be instrumental in quantifying how often a random variable X deviates from the expectation $\mathbb{E}[X]$. Its definition reads:

$$\mathsf{Var}(X) := \mathbb{E}[(X-\mu)^2] \tag{1}$$

where $\mu := \mathbb{E}[X]$. We next learned about a useful fact which often eases the variance calculation:

$$\mathsf{Var}(X) = \mathbb{E}[X^2] - \mu^2. \tag{2}$$

We also investigated two properties w.r.t. *independent* random variables X and Y:

$$\begin{aligned} \text{(Uncorrelatedness): } & \mathbb{E}[XY] = \mathbb{E}[X]\mathbb{E}[Y]; \\ \text{(Additivity): } & \mathsf{Var}(X+Y) = \mathsf{Var}(X) + \mathsf{Var}(Y). \end{aligned} \tag{3}$$

So far we have considered only one type of random variables in which we take *discrete*-valued real numbers. In reality, however, there are many scenarios concerning *continuous*-valued random quantities. One such important scenario is *communication.* In the context of communication, *noise* is added into the system and it can be mathematically modeled as a continuous-valued random quantity. At the end of the last lecture, I mentioned that there is a relevant concept concerning such continuous random signal. That is, *continuous* random variable.

Today's lecture

Today we will investigate the continuous random variable. Specifically what we are going to cover are four folded. We will first introduce the concept of a continuous random variable in the context of a new type of sample space that we did not explore. We will then study a new concept called the *probability density function* that arises in the context of a continuous random variable. Next we will study a relevant concept named the *cumulative density function.* Finally we will investigate the definitions of expectation and variance w.r.t. a continuous random variable.

A new type of sample space

So far we have considered a special type of sample space which is a *countable* set, i.e., either a *finite* set or a *countably infinite* set. In real life, however, there are tons of situations where a sample space is not countable. To see this clearly, consider an experiment of picking up a point in the $[0,1]$ interval, uniformly at random. See Fig. 1. Notice that an element ω in Ω is a *continuous* value. So the sample space is *not countable.* A *continuous* random variable can easily come up in such a context. One natural *continuous* random variable that one can think of is a mapping that just outputs the fed element: $X(\omega) = \omega$. From this, one can readily image what a continuous random variable means. Its definition is exactly the same as that of the discrete counterpart, except that it takes a *continuous* value in the range.

How to define the probability distribution?

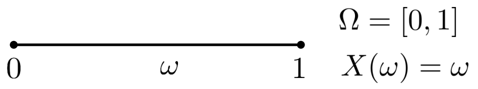


Figure 1: A new type of sample space: An element in Ω is a *continuous* value that lies in between 0 and 1. Here $X(\omega) = \omega$ is the simplest *continuous* random variable that one can naturally think of.

For such continuous random variable, we should also worry about how to define its probability distribution. One natural choice might be the discrete counterpart: $\mathbb{P}_X(x) = \mathbb{P}(X = x)$. But there is an issue in this naive choice. The issue is that in this case, we should set $\mathbb{P}(X = x) = 0$ for all $x \in \mathcal{X}$, no matter how X behaves. Why? Suppose we assign some tiny yet positive value ϵ to $\mathbb{P}(X = x) = \epsilon$ for all $x \in \mathcal{X}$. By uniformity this probability should be the same for all x. But then, the sum of all probabilities $\mathbb{P}(X = x)$ will be ∞, violating the sum-up-to-one constraint. Hence, $\mathbb{P}(X = x)$ must be zero for all $x \in \mathcal{X}$. Obviously it is not a proper way to go.

To avoid such uninteresting situations, we may consider the probability of X being now in an *interval*:

$$\mathbb{P}(X \in [a,b]) \tag{4}$$

where $0 \leq a \leq b \leq 1$. In this case, the natural probability assignment would be to take the length of the associated interval $[a,b]$ normalized by the entire interval $[0,1]$:

$$\mathbb{P}(X \in [a,b]) = \frac{\text{length of } [a,b]}{\text{length of } [0,1]} = b - a. \tag{5}$$

This is indeed a proper choice. Why? The probability increases with the length of the associated interval. So it makes an intuitive sense. It also satisfies the sum-up-to-one constraint that the probability distribution of a discrete random variable respects. To see this, consider *disjoint* intervals I_i's such that $\bigcup_i I_i = [0,1]$. Then,

$$\sum_i \mathbb{P}(X \in I_i) = \mathbb{P}\left(\bigcup_i I_i\right) = 1 \tag{6}$$

where the 1st equality follows from the fact that I_i's are disjoint.

Probability density function

Now what about for general cases, not necessarily such uniform distribution case? In light of (4), for generality, we need to specify $\mathbb{P}(X \in [a,b])$ for all intervals $[a,b]$, not limited to the $[0,1]$ interval. Also, $\mathbb{P}(X \in [a,b])$ may not be a sole function of the interval length $b - a$. It may also depend on how often X belongs to the interval. This is where the concept of a *probability density function* (pdf for short) kicks in. It turns out that the pdf serves to formally specify the probability distribution. A pdf is denoted by $f(x)$ and is defined as a function that satisfies:

$$\mathbb{P}(a \leq X \leq b) = \int_a^b f(x)dx \qquad \forall a, b \in \mathbb{R} \text{ such that } a \leq b \tag{7}$$

where $f(x)$ is assumed to be continuous everywhere, i.e., integrable. Notice that this definition allows us to specify the interested probabilities $\mathbb{P}(a \leq X \leq b)$ with the pdf $f(x)$. Someone may want to use a different notation like $f_X(x)$ to highlight an associated random variable X. But many people including me use a simpler notation $f(x)$ as above. Here we have *integration* of

$f(x)$. So it has a geometric interpretation: the area under the function $f(x)$ spanned by an associated interval. So the pictorial meaning of $\mathbb{P}(a \le X \le b)$ in (7) is the *area* below the pdf in the associated interval $[a, b]$, as illustrated in Fig. 2. So we see that the pdf plays a similar role as the *histogram* that was used for illustration of the probability distribution w.r.t. a *discrete* random variable.

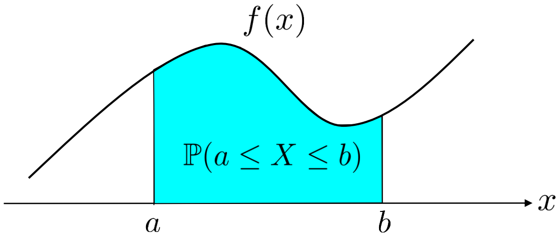


Figure 2: Pictorial illustration of a probability density function $f(x)$ that shows the relationship with $\mathbb{P}(a \le X \le b)$.

Two properties of the pdf

Similar to the histogram in the discrete counterpart, there are two properties w.r.t. the pdf. The first is the non-negativity property:

$$f(x) \ge 0. \tag{8}$$

This is due to the definition of the pdf (7). Suppose $f(x) < 0$ for some value, say t. Then, one can find an interval that includes t so that the integral over the interval in (7) is negative. So we would have a negative probability for such an event, which violates the non-negativity property of the probability distribution. The second is the sum-up-to-one property:

$$\int_{-\infty}^{\infty} f(x)dx = \mathbb{P}(-\infty \le X \le \infty) = 1 \tag{9}$$

where the 1st equality is due to the definition of the pdf (7); and the 2nd equality is because X must take on some value in the real line $\mathbb{R}$.

Two caveats

However, there are significant distinctions w.r.t. the histogram. These are reflected in the following two caveats. The first is that the integration in (7) may not be well defined. But this is a very rare and practically-irrelevant case. So we will not worry about this throughout the course. Actually forgetting about this is indeed okay unless you wish to do something related to the hardcore probability theory in the future. I assume you may not be interested in the hardcore probability, as most of you guys are from engineering-related departments. If you do so, then you may want to take a graduate-level hardcore probability course, e.g., "measure theory".

The second caveat is that the pdf is *not a probability quantity.* This can easily been seen from the following example. Consider a uniformly distributed random variable X that takes a continuous value in $[0, 0.5]$. In this case, $f(x) = 2$ due to (9). Since the pdf $f(x) = 2$ exceeds 1, it is obviously not a probability quantity. You may then ask: What does the pdf mean? How is the pdf related to the probability quantity? To answer this, consider a very small interval, say

$[x, x+\delta]$, wherein one can approximate:

$$\mathbb{P}(x \leq X \leq x+\delta) = \int_x^{x+\delta} f(x)dx \approx \delta f(x).$$

This approximation becomes more accurate as $\delta \rightarrow 0$, as the integration becomes closer to the area of the rectangle with width δ and height $f(x)$. More formally, in the limit of $\delta \rightarrow 0$, we get:

$$f(x) = \lim_{\delta \rightarrow 0} \frac{\mathbb{P}(x \leq X \leq x+\delta)}{\delta}. \tag{10}$$

Hence, the pdf can be interpreted as the *probability per unit length*. Actually this is the very reason that people use the word "density" in the naming.

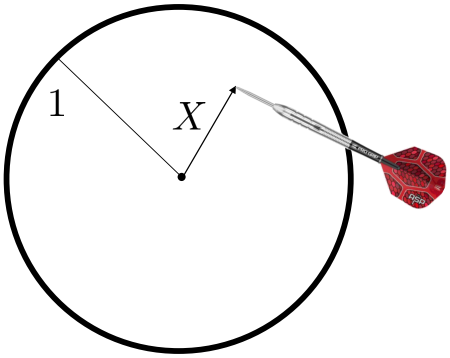


Figure 3: Experiment of throwing darts. Let X be a continuous random variable that indicates a distance from the center to the place that the dart lands in.

Example: Throwing darts

Let's do some exercise on how to compute $f(x)$ via an example. Consider an experiment of throwing a dart into a circle-shaped target with a unit radius. See Fig. 3. Suppose we are interested in the distance from the center to the location that the thrown dart points to. Denote the distance by a continuous random variable X. The range of a continuous random variable is always the real line $\mathcal{X} = [-\infty, \infty]$. But in this experiment, X cannot be negative. So a reasonable assumption is $\mathbb{P}(X < 0) = 0$. Also, let us ignore the case where the dart is out of the target circle, so we assume $\mathbb{P}(X > 1) = 0$.

Now how to compute the probability density function $f(x)$ in the interested range $x \in [0, 1]$? To this end, we first need to compute $\mathbb{P}(x \leq X \leq x+\delta)$ in light of (10). This probability must be proportional to the area of the ring squeezed between the x-radius circle and the $(x+\delta)$-radius circle. See the green-colored ring in Fig. 4. Hence, it should read:

$$\begin{aligned}\mathbb{P}(x \leq X \leq x+\delta) &= \frac{\text{ring area}}{\pi(1)^2} \\ &= \frac{\pi(x+\delta)^2 - \pi x^2}{\pi} \\ &= 2\delta x + \delta^2\end{aligned}$$

where the 1st equality is due to the normalization by the unit-circle area $\pi(1)^2 = \pi$. Why normalized? This is because of the sum-up-to-one constraint. This together with (10) and the

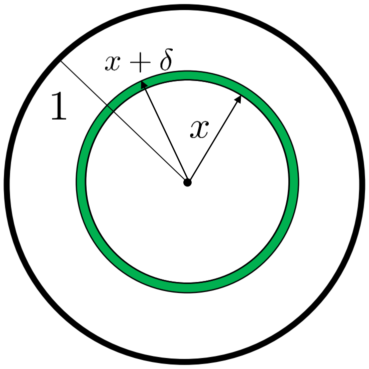


Figure 4: The probability $\mathbb{P}(x \leq X \leq x+\delta)$ corresponds to the area of the green-colored ring placed in between the x-radius circle and the $(x+\delta)$-radius circle.

assumption $\mathbb{P}(X > 1) = \mathbb{P}(X < 0) = 0$ gives:

$$f(x) = \begin{cases} 0, & \text{if } x < 0; \\ 2x, & \text{if } 0 \leq x \leq 1; \\ 0, & \text{if } x > 1. \end{cases} \tag{11}$$

If you think about it, reading $f(x) = 2x$ in the interested range makes an intuitive sense. Why? A ring farther away from the center has a larger area than a ring closer to the center with the same width δ.

Cumulative density function

In the discrete random variable case, the histogram fully specifies the statistical behaviour of a random variable. So you may guess that only the pdf (playing a similar role as the histogram) is needed in the context of continuous random variables. But it turns out the story is a bit different in the continuous case. We should worry about another relevant concept. To see this, first recall the relationship between the pdf and the probability measure:

$$f(x) = \lim_{\delta \to 0} \frac{\mathbb{P}(x \leq X \leq x+\delta)}{\delta}. \tag{12}$$

Here a key observation that one can make is:

$$\begin{aligned} \mathbb{P}(x \leq X \leq x+\delta) &= \mathbb{P}(X \leq x+\delta) - \mathbb{P}(X < x) \\ &= \mathbb{P}(X \leq x+\delta) - \mathbb{P}(X \leq x) \end{aligned} \tag{13}$$

where the 1st equality is because the event $\{X \leq x+\delta\}$ can be decomposed into the two *disjoint* events $\{X < x\}$ and $\{x \leq X \leq x+\delta\}$; and the 2nd equality follows from $\mathbb{P}(X = x) = 0$ for a continuous random variable X. Defining $F(x) := \mathbb{P}(X \leq x)$, we can then rewrite (12) as:

$$f(x) = \lim_{\delta \to 0} \frac{F(x+\delta) - F(x)}{\delta}.$$

What does this remind you of? Yes, it is the *derivative* $\frac{d}{dx}F(x)$! Hence, we get:

$$f(x) = \frac{d}{dx}F(x). \tag{14}$$

Taking integration on both sides, we also obtain another equivalent representation:

$$F(x) = \int_{-\infty}^{x} f(t)dt. \tag{15}$$

Here $F(x)$ is an important function that we need to worry about in the context of continuous random variables. Prior to talking about its importance, let us interpret what $F(x)$ means from (15). Pictorially, $F(x)$ indicates the area under $f(x)$ spanned by the interval $[-\infty, x]$. See Fig. 5. So $F(x)$ can be interpreted as the area *accumulated* up to x. Hence, it is called the *cumulative density function*, cdf for short.

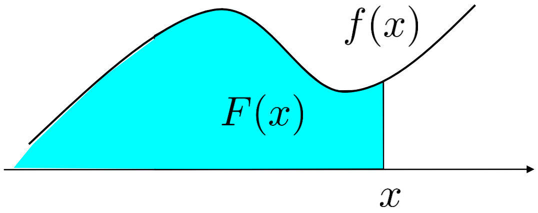


Figure 5: Illustration of the cumulative density function, usually denoted by $F(x)$. The pictorial meaning of $F(x)$ is the area below $f(x)$ from $-\infty$ up to x.

Importance of the cdf

Pondering

$$f(x) = \frac{d}{dx}F(x), \qquad F(x) = \int_{-\infty}^{x} f(t)dt, \tag{16}$$

we see that the pdf and cdf contain exactly the same information: one implies the other and vice versa. Why? $f(x) = \frac{d}{dx}F(x)$ implies $F(x) = \int_{-\infty}^{x} f(t)dt + C$. But C should be zero due to $F(\infty) := \mathbb{P}(X \leq \infty) = 1 = \int_{-\infty}^{\infty} f(t)dt$. The other direction is straightforward.

Now you may wonder why we care about the concept of cdf? Isn't the pdf enough? The reason is that it turns out for many problems, the cdf is much easier to compute relative to the pdf. Hence, usually we first compute the cdf and then compute the pdf from the cdf. Like the pdf, the cdf has similar properties:

1. (Non-negativity): $0 \leq F(x) \leq 1$;
2. (Terminal point): $F(\infty) = \int_{-\infty}^{\infty} f(t)dt = 1$;
3. (Initial point): $F(-\infty) = 0$.

Expectation and variance

For a continuous random variable X, the expectation is defined as:

$$\mathbb{E}[X] := \int_{-\infty}^{\infty} x f(x)dx. \tag{17}$$

The rationale behind the definition is the same as the discrete-case counterpart. Here, $f(x)dx$ acts as a proper weight that captures the frequency of the occurrence of x, as $\mathbb{P}_X(x)$ does in the discrete case. Similarly we define the variance as:

$$\mathsf{Var}(X) := \mathbb{E}[(X-\mu)^2] \tag{18}$$

where $\mu := \mathbb{E}[X]$. We also have a useful fact as in the discrete case:

$$\begin{aligned}\mathsf{Var}(X) &= \mathbb{E}[(X-\mu)^2] \\ &= \mathbb{E}[X^2] - \mu^2 \\ &= \int_{-\infty}^{\infty} x^2 f(x) dx - \mu^2\end{aligned}$$

where the 2nd equality is due to the linearity of expectation.

Look ahead

As I mentioned earlier, one of the main reasons that we study continuous random variables is in their relevancy to the noise in communication systems that we will delve into in Part III: Applications. It turns out the noise signal can be modeled as a very popular continuous random variable. That is, the *Gaussian* random variable. So next time, we will investigate the Gaussian random variable.

Lecture 11: Gaussian random variables

Recap

Last time we have studied another type of random variables: *continuous* random variables. Like a *discrete* random variable, it is also a function that maps an element in Ω into a real-valued number. But the key distinction is that it takes a *continuous* value in the range, while the discrete counterpart takes a concretely specified value called the *discrete* (or *countable*) value. Since the natural probability assignment for an event of a continuous random variable taking a particular value exactly is zero, a proper probability distribution of interest is the probability that it belongs to an interested *interval.* So in this context, we need to specify $\mathbb{P}(a \leq X \leq b)$ all admissible intervals $[a, b]$. In order to formally specify such probabilities, a new concept that plays a similar role as the histogram in the discrete counterpart has been introduced. That is, the probability density function (pdf) $f(x)$, defined as a function that satisfies:

$$\mathbb{P}(a \leq X \leq b) = \int_a^b f(x)dx \qquad \forall a, b \in \mathbb{R} \text{ such that } a \leq b. \tag{1}$$

The word "density" in the naming comes from the fact that it can be interpreted as the *probability per unit length* in light of another expression: $f(x) = \lim_{\delta \to 0} \frac{\mathbb{P}(x \leq X \leq x+\delta)}{\delta}$. We have also introduced another seemingly-redundant concept: the cumulative density function (cdf) $F(x) := \mathbb{P}(X \leq x)$, which has the the following relationship with $f(x)$:

$$f(x) = \frac{d}{dx}F(x), \;\; F(x) = \int_{-\infty}^{x} f(t)dt. \tag{2}$$

The rationale behind the introduction of the cdf is that in many problems, the cdf is much easier to compute; hence people often try to compute the cdf and then obtain the pdf accordingly.

Like I said earlier, there is one very popular continuous random variable that is quite instrumental in many applications: the *Gaussian* random variable. As a prominent example, I mentioned that a *noise* signal in communication systems of one focused topic in Part III can be modeled as a Gaussian random variable.

Today's lecture

Today we will investigate details on the Gaussian random variable. This lecture is comprised of four parts. As usual, we will start by introducing the definition of the Gaussian random variable. As you will see soon, it is defined via a particular form of a pdf. In the second part, we will verify that the pdf indeed satisfies the sum-up-to-one constraint, and we will also compute its mean and variance. Next we will study one important property, called *normality preservation*, which says that any *linear transform* of a Gaussian random variable is still Gaussian. It turns out this property enables us to compute the cdf of any Gaussian random variable efficiently. So in the last part, we will study how to compute the cdf of a Gaussian random variable with the property.

Definition of a Gaussian random variable

We say that a random variable X is *Gaussian* if its pdf reads:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \qquad \forall x \in \mathbb{R} \tag{3}$$

where μ and $\sigma^2 > 0$ are some constants which are shown to be mean and variance, respectively (this will be verified soon). We ignore the trivial case $\sigma^2 = 0$ that leads to a *deterministic* X. This will be clearer soon. As I mentioned earlier, it often (normally) appears in a wide variety of scenarios. Hence, it is also called the *normal* random variable. Here the function $f(x)$ is named the Gaussian (or normal) distribution.

Why Gaussian random variables become popular?

The reason can be explained via the following two sequential facts. The first is that in many scenarios (like communication), a signal of interest (like a noise signal) is shown to be expressed as the sum of many independent random variables. Second, there is a prominent theorem which says that the sum with a proper scaling can be approximated as a Gaussian random variable as the number of involved variables in the summation grows large. The theorem is the very famous *Central Limit Theorem (CLT)* that you may hear of. This is the key reason behind the popularity of the Gaussian distribution. We will discuss more details later particularly in Part III where we will talk about the communication application.

Sum-up-to-one constraint

As promised in the beginning, let us verify that the Gaussian distribution (3) indeed satisfies the sum-up-to-one constraint:

$$\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx = 1. \tag{4}$$

Since the pdf form is a bit dirty involving parameters like μ and σ^2, we first simplify it via a well-known technique called the *change of variable*. Taking $t = \frac{x-\mu}{\sigma}$, we get $dt = \frac{dx}{\sigma}$. Putting all these into the LHS in the above, we obtain:

$$\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt. \tag{5}$$

Although we did simplification, we feel a bit stuck because there is no concrete integral formula for $e^{-t^2/2}$. Now is the moment that you need to exploit your knowledge and experience obtained from Calculus: *converting into Polar coordinate*. To understand what it means, first observe that the integral in (5) is positive due to the positive pdf. Hence, it suffices to prove the square of the integral is 1:

$$\left(\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt \right)^2 = 1.$$

By using two different dummy variables (say x and y) in the double integral, we can rewrite the LHS in the above as:

$$\begin{aligned} \left(\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt \right)^2 &= \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-\frac{x^2}{2}} dx \int_{-\infty}^{\infty} e^{-\frac{y^2}{2}} dy \\ &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-\frac{x^2+y^2}{2}} dx dy \end{aligned}$$

This is where the Polar coordinate (r, θ) kicks in:

$$x = r\cos\theta, \;\; y = r\sin\theta.$$

Using the fact that $(\sin\theta)^2 + (\cos\theta)^2 = 1$, we get $x^2 + y^2 = r^2$. Now how to express $dxdy$ in terms of $(dr, d\theta)$? By visualizing the area change due to transition from (r, θ) into $(r + dr, \theta + d\theta)$, we can figure out:

$$\begin{aligned} \text{area change} &\approx dr \times (\text{width change due to } d\theta) \\ &= dr \times (rd\theta) = rdrd\theta. \end{aligned} \tag{6}$$

Hence, the area change $dxdy$ in the Cartesian coordinate can be translated into that of the Polar coordinate: $rdrd\theta$. This together with $x^2 + y^2 = r^2$ yields:

$$\begin{aligned} \left(\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt \right)^2 &= \frac{1}{2\pi} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} e^{-\frac{x^2+y^2}{2}} dxdy \\ &= \frac{1}{2\pi} \int_0^{2\pi} \int_0^{\infty} e^{-\frac{r^2}{2}} rdrd\theta \\ &= \int_0^{\infty} e^{-\frac{r^2}{2}} rdr \\ &= \int_0^{\infty} e^{-u} du \\ &= \left[-e^{-u}\right]_0^{\infty} = 1 \end{aligned}$$

where the 3rd equality is due to the integration over θ from 0 to 2π; and the 2nd last equality follows from the change of variable: $u = \frac{r^2}{2}$ and $du = rdr$. This completes the proof of (4).

Computation of $\mathbb{E}[X]$ and $\mathsf{Var}(X)$

The expectation calculation is straightforward. By linearity, we first get:

$$\mathbb{E}[X - \mu] = \mathbb{E}[X] - \mu. \tag{7}$$

We then manipulate $\mathbb{E}[X - \mu]$ as:

$$\begin{aligned} \mathbb{E}[X - \mu] &= \int_{-\infty}^{\infty} (x - \mu) \cdot \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx \\ &= \int_{-\infty}^{\infty} t \cdot \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{t^2}{2\sigma^2}} dt \\ &= 0 \end{aligned}$$

where the 2nd equality is due to the change of variable $t = x - \mu$; and the last equality is because the pdf (marked in blue) is symmetric around $t = 0$ and hence the interested function multiplied by t is an *odd* function. Applying this into (7), we obtain the expectation as:

$$\mathbb{E}[X] = \mu.$$

Next let us calculate the variance. The variance computation is a bit tricky. Using the definition of $\mathsf{Var}(X)$ together with the change of variable $t = \frac{x-\mu}{\sigma}$, we get:

$$\begin{aligned} \mathsf{Var}(X) &:= \int_{-\infty}^{\infty} (x - \mu)^2 \cdot \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx \\ &= \frac{\sigma^2}{\sqrt{2\pi}} \int_{-\infty}^{\infty} t^2 e^{-\frac{t^2}{2}} dt. \end{aligned}$$

Now how to integrate $t^2 e^{-t^2/2}$? Again, you may feel headache because there is no concrete integral formula for the function. This is where another well-known technique that you learned from Calculus comes in. That is, *integration by parts*:

$$\int f \cdot g' = fg - \int f' \cdot g. \tag{8}$$

Defining $f = -t$ and $g' = -te^{-t^2/2}$, and applying the "integration by parts" (8), we get:

$$\begin{aligned}
\mathsf{Var}(X) &:= \int_{-\infty}^{\infty} (x-\mu)^2 \cdot \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} dx \\
&= \frac{\sigma^2}{\sqrt{2\pi}} \int_{-\infty}^{\infty} t^2 e^{-\frac{t^2}{2}} dt \\
&= \frac{\sigma^2}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \underbrace{-t}_{f} \cdot \underbrace{\left(-te^{-\frac{t^2}{2}}\right)}_{g'} dt \\
&= \frac{\sigma^2}{\sqrt{2\pi}} \left[-t \cdot e^{-\frac{t^2}{2}}\right]_{-\infty}^{\infty} - \frac{\sigma^2}{\sqrt{2\pi}} \int_{-\infty}^{\infty} (-1) \cdot e^{-\frac{t^2}{2}} dt \\
&= \sigma^2 \cdot \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-\frac{t^2}{2}} dt \\
&= \sigma^2
\end{aligned}$$

where the last equality is due to the sum-up-to-one constraint of the Gaussian distribution (marked in blue). Since $\mathbb{E}[X] = \mu$ and $\mathsf{Var}(X) = \sigma^2$, the Gaussian distribution looks like the one in Fig. 1. It is sort of "bell-shaped", centered at (and symmetric around) $x = \mu$, and "width" determined by the standard deviation σ.

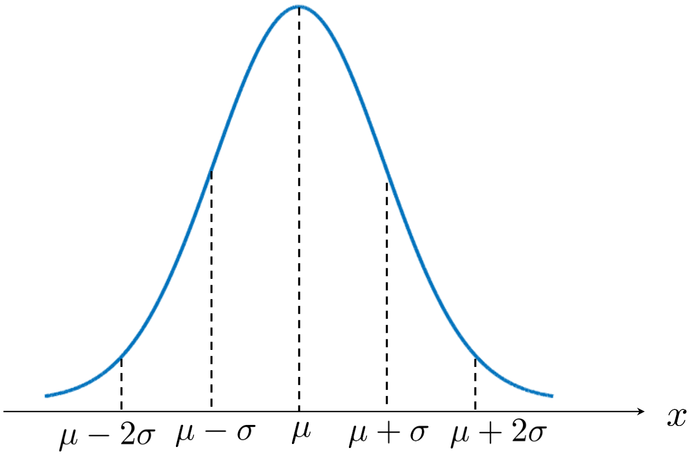


Figure 1: The Gaussian distribution with mean μ and variance σ^2.

Normality preservation property

So far we have done some boring stuffs that require complicated-looking integrations. Let us now switch gears to touch upon some exciting and very useful stuffs. One such stuff is a very important property called "normality preservation". Let X be a Gaussian random variable with mean μ and variance σ^2. It is simply denoted by $X \sim \mathcal{N}(\mu, \sigma^2)$ where $\mathcal{N}$ stands for "Normal". The normality preservation property says that any *linear transformation* of X also respects the normal distribution: for any constants (c_1, c_2),

$$Y = c_1 X + c_2 \sim \mathcal{N}(c_1\mu + c_2, c_1^2\sigma^2). \tag{9}$$

The proof of this is as follows. The case $c_1 = 0$ is trivial. In this case, $Y = c_2$. First consider the case $c_1 > 0$. Using the definition of the cdf, we get:

$$\begin{aligned} F_Y(y) &:= \mathbb{P}(Y \leq y) \\ &= \mathbb{P}(c_1 X + c_2 \leq y) \\ &= \mathbb{P}\left(X \leq \frac{y - c_2}{c_1}\right) \\ &= F_X\left(\frac{y - c_2}{c_1}\right) \end{aligned}$$

where the 3rd equality is due to $c_1 > 0$. Taking derivatives w.r.t. y on both sides, we obtain:

$$\begin{aligned} \text{Case } c_1 > 0: \; f_Y(y) &= \frac{d}{dy} F_X\left(\frac{y - c_2}{c_1}\right) \\ &= \frac{d}{dx} F_X\left(\frac{y - c_2}{c_1}\right) \cdot \frac{d}{dy}\left(\frac{y - c_2}{c_1}\right) \\ &= \frac{1}{c_1} f_X\left(\frac{y - c_2}{c_1}\right) \\ &= \frac{1}{c_1} \cdot \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{\left(\frac{y-c_2}{c_1} - \mu\right)^2}{2\sigma^2}} \\ &= \frac{1}{\sqrt{2\pi} c_1 \sigma} e^{-\frac{(y - (c_1\mu + c_2))^2}{2c_1^2\sigma^2}} \end{aligned} \tag{10}$$

where the 2nd equality is due to the chain rule ($\frac{d}{dy} g(x) = \frac{d}{dx} g(x) \cdot \frac{dx}{dy}$); and the 2nd last equality follows from the formula of the Gaussian distribution (3). The case $c_1 < 0$ is similar. In this case, one can readily verify that:

$$\text{Case } c_1 < 0: \; f_Y(y) = -1 \cdot \frac{1}{\sqrt{2\pi} c_1 \sigma} e^{-\frac{(y - (c_1\mu + c_2))^2}{2c_1^2\sigma^2}}. \tag{11}$$

If you are not convinced, please check this. This together with (10) gives:

$$Y \sim \mathcal{N}(c_1\mu + c_2, c_1^2\sigma^2). \tag{12}$$

We can also double-check in part via the following calculations:

$$\begin{aligned} \mathbb{E}[Y] &= c_1 \mathbb{E}[X] + c_2 = c_1\mu + c_2; \\ \mathsf{Var}(Y) &:= \mathbb{E}[(Y - (c_1\mu + c_2))^2] \\ &= \mathbb{E}[c_1^2 (X - \mu)^2] \\ &= c_1^2 \mathbb{E}[(X - \mu)^2] = c_1^2\sigma^2. \end{aligned} \tag{13}$$

Computation of the cdf of a Gaussian random variable

One significant consequence of the normality preservation property (9) is that any Gaussian random variable, say $X \sim \mathcal{N}(\mu, \sigma^2)$, can be represented as a linear transformation of the *standard Gaussian* random variable with mean 0 and variance 1, say $Z \sim \mathcal{N}(0,1)$:

$$X = \sigma Z + \mu. \tag{14}$$

It turns out this property (14) enables us to compute the cdf of any Gaussian random variable efficiently. To see this, first observe the cdf of $X \sim \mathcal{N}(\mu, \sigma^2)$:

$$\begin{aligned} F_X(x) &= \mathbb{P}(X \leq x) \\ &= \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(t-\mu)^2}{2\sigma^2}} dt. \end{aligned} \tag{15}$$

Here we are faced with a sort of challenge, as it requires the computation of integration which has *no closed form solution*. This is where the property (14) plays a role. Using the property, we can convert the integration into a function of *tractable* integration associated with the standard Gaussian $\mathcal{N}(0,1)$. The good thing about $\mathcal{N}(0,1)$ is that we can obtain a numerical value of such integration although we do not have a closed formula. See below for details:

$$\begin{aligned} F_X(x) &= \mathbb{P}(X \leq x) \\ &= \mathbb{P}\left(\frac{X-\mu}{\sigma} \leq \frac{x-\mu}{\sigma}\right) \\ &= \int_{-\infty}^{\frac{x-\mu}{\sigma}} \frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}} dz =: \Phi\left(\frac{x-\mu}{\sigma}\right) \end{aligned} \tag{16}$$

where $Z = \frac{X-\mu}{\sigma} \sim \mathcal{N}(0,1)$. Here the cdf of the standard Gaussian distribution is denoted by $\Phi(z) := \mathbb{P}(Z \leq z)$. As mentioned earlier, it has already be computed numerically and tabulated in many places including many books and Wikipedia.

Numerical values of $\Phi(z)$ are even accessible in Python via a function named "scipy.stats.norm.cdf":

$$\mathsf{norm.cdf}(z) = \Phi(z). \tag{17}$$

Applying this into (16), we get:

$$F_X(x) = \Phi\left(\frac{x-\mu}{\sigma}\right) = \mathsf{norm.cdf}\left(\frac{x-\mu}{\sigma}\right). \tag{18}$$

Look ahead

We have thus far studied numerous concepts, laws and techniques: sample space, probability model, events, conditional probability, total probability law, Bayes' law, independence, union bound, discrete random variables, pmf, expectation, variance, Chebyshev's inequality, Markov's inequality, continuous random variables, pdf, cdf, Gaussian random variables. These form the contents of Part I.

Unfortunately, there is one more concept required to understand the two key principles (MAP and ML estimation) that we aimed at in Lecture 1. That is, the concept of *random processes*. So next time, we will investigate random processes. This is the beginning of Part II. For the rest of Part II, we will study the two key principles (finally!) together with relevant important theorems: the Law of Large Numbers and the Central Limit Theorem.

EE210 Probability and Introductory Random Processes April 8, 2021
KAIST, Spring 2021 Changho Suh (chsuh@kaist.ac.kr)

Lecture 12: Random processes

Recap

In Part I, we have studied various fundamental stuffs in probability. These include: (i) numerous concepts like sample space, events, conditional probability, independence, random variables, pmf, pdf and cdf; (ii) important laws such as total probability law, Bayes' law and De Morgan's law; and (iii) useful techniques like the complement counting trick, the union bound, Markov's inequality and Chebyshev's inequality.

In Part II, we will attack the two key principles that we targeted in the first lecture: MAP and ML estimation. But prior to this, we need to learn about one more important concept: *random processes*.

Today's lecture

Today we will embark on Part II, exploring the first topic: random processes. Specifically what we are going to cover are three folded. First of all, we will introduce the definition of random processes, together with the rationale behind the definition as usual. It turns out many interested signals in reality can be modeled as one particular type of random processes, named the *stationary process*. So in the second part, we will study the stationary process together with its special yet very famous version, called the *i.i.d. process*. Lastly we will investigate three prominent examples (of such stationary process class) which are shown to be practically relevant, as well as which we will study in depth about in Part III: Applications. These are: (i) *Bernoulli* process; (ii) *Gaussian* process; and (iii) *Markov* process.

Definition of a random process

The definition is extremely simple to state. A random process is simply a *sequence of random variables*. That's it – period! Two useful remarks though. First, it is mostly a *time-series* sequence, i.e., the associated random variables often evolve in time. Second, the sequence can be either finite or *infinite*. Hence, it is denoted by $(X_1, X_2, \ldots, X_n)$ or $(X_1, X_2, \ldots, X_n, \ldots)$. The simpler notation is: $\{X_i\}_{i=1}^n$ or $\{X_i\}_{i=1}^\infty$. Let $X_i \in \mathcal{X}_i$ for all i. Then, the range reads:

$$\mathcal{X} := \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_n \times \cdots$$

In general, we have two type of notations for the probability distribution depending on whether X_i is discrete or continuous. For simplicity, we ignore a rare yet possible mixed scenario in which some are discrete while the others are continuous. For the discrete case, we use the thick $\mathbb{P}$ notation. For all finite subsets of $\{x_i\}_{i=1}^\infty$, e.g., for $\{x_i\}_{i=1}^n$:

$$\mathbb{P}(x_1, x_2, \ldots, x_n). \tag{1}$$

To distinguish it from the probability distribution w.r.t. a sample space Ω, one may prefer the following more formal notation:

$$\mathbb{P}_{X_1,\ldots,X_n}(x_1, x_2, \ldots, x_n).$$

But this notation looks complicated, as it puts lots of stuffs in the subscript. Hence, people like me prefer the simpler notation (1). Since it involves many random variables, it is also called the *joint distribution.* On the other hand, for the continuous case, we employ the f notation. For all finite subsets of $\{x_i\}_{i=1}^{\infty}$, e.g., for $\{x_i\}_{i=1}^{n}$:

$$f(x_1, x_2, \ldots, x_n). \tag{2}$$

Applications of random processes

Why do we care about random processes? Obviously it is because there are many applications. In Lecture 1, I put a particular emphasis on the following three killer applications: (i) communication; (ii) speech recognition; (iii) machine learning. In communication systems, *noise signals* that evolve in time can be modeled as a random process, more precisely, the *Gaussian* process that we will figure out soon. In speech recognition, *voice signals* that are fed as an input to the system can be modeled as a random process, more precisely, the *Markov* process that we will also investigate soon. In machine learning, data samples, e.g., input samples which we denoted by $\{x^{(i)}\}_{i=1}^{m}$, can be interpreted as a random process, more precisely, the *i.i.d.* process that we will study as well. Of course, that's just the tip of the iceberg. There are tons of instances in reality that can be modeled as a random process, like a sequence of daily stock prices; a sequence of daily temperatures measured every morning; a sequence of call inter-arrival times.

Two important types of random processes

There is a particular type of random processes which is quite instrumental in modeling many signals of interest in reality. That is, the *stationary* process. The definition is also simple to state. We say that $(X_1, X_2, \ldots, X_n, \ldots)$ is *stationary* if its shifted version $(X_{1+\ell}, X_{2+\ell}, \ldots, X_{n+\ell}, \ldots)$ is a *statistical copy* of the original sequence for all shifts ℓ's. Here what it means by "statistical copy" is having the *same joint distribution.*

There is another much simpler process, named the *i.i.d.* process. It is a special yet very famous version of the stationary process. We say that $(X_1, X_2, \ldots, X_n, \ldots)$ is *i.i.d.* if the associated random variables are mutually independent, and each has the same distribution (identically distributed). You may be able to figure out what the i.i.d. means. Yes, it stands for "$\underline{\text{i}}$ndependent & $\underline{\text{i}}$dentically $\underline{\text{d}}$istributed". Here the mutual independence for an infinite number of random variables means that for *all finite subsets* of the random variables. Notice that the i.i.d. process is obviously stationary because its shifted version has the same joint distribution.

Example #1: Bernoulli process

Below we list three prominent examples which belong to either the i.i.d. process or the stationary process. The first is a very simple and famous one, called the *Bernoulli process.* It is named after one of the math heroes in history, Jacob Bernoulli. See Fig. 1. We say that $(X_1, X_2, \ldots, X_n, \ldots)$ is the Bernoulli process if X_i's are i.i.d. and each is binary, i.e., $X_i \in \{0, 1\}$, with $\mathbb{P}(X_i = 1) = p$. Each random variable is simply denoted by $X_i \sim \mathsf{Bern}(p)$. In fact, Jacob Bernoulli employed such a simple process in the course of discovering one of the foundational laws in mathematics, called the Law of Large Numbers (LLN). Due to the importance of the great discovery, people named the employed random process the Bernoulli process. Later we will have a chance to study the LLN in depth. Please be patient until we get to the point.

As you may figure out, we already saw the Bernoulli process in past lectures. Remember the experiment of flipping a p-biased coin n times where p indicates the probability of each flip showing "Head". In the experiment, the basic random variables that we defined were:

$$X_i = \mathbf{1}\{i\text{th flip shows "Head"}\} \qquad \forall i \in \{1, 2, \ldots, n\}.$$

Figure 1: Picture of Jacob Bernoulli. He is a Swiss mathematician in the 1600s. He is very famous for discovering the number e and one of the foundational laws in mathematics, called the Law of Large Numbers (LLN).

This is indeed the Bernoulli process.

The Bernoulli process also forms the basis of many other interested random variables. One such random variable is the total number S of "Head"s out of n flips. Again, this is also what we investigated before. Remember that its probability distribution is binomial:

$$\mathbb{P}_S(s) = \binom{n}{s} p^s (1-p)^{n-s} \qquad \forall s \in \{0, 1, \ldots, n\}.$$

Another random variable is the total number X of flips until we see "Head". Again, this is also what you saw, not in class, but in PS3. Yes, it is the *geometric distribution*:

$$\mathbb{P}_X(x) = (1-p)^{x-1} p \qquad \forall x \in \{1, 2, 3, \ldots\}.$$

An application of the Bernoulli process

You may feel bored about the flipping-coin example. The Bernoulli process is useful also in more practically-relevant settings. One such application is the *call arrivals problem.* See Fig. 2. Suppose the entire time window is discretized into n small intervals and each interval receives only one call if any, i.e., only two events can happen: either no call initiation is made or exactly one. We also assume that each interval has a call arrival with probability p, and call arrivals are independent from interval to interval. Let

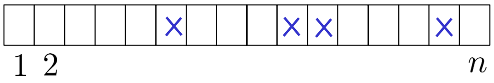


Figure 2: Call arrivals problem.

$$X_i = \mathbf{1}\left\{i\text{th interval has a call arrival}\right\}. \tag{3}$$

Then, we can immediately see that $\{X_i\}_{i=1}^n$ is the Bernoulli process, each being distributed according to $X_i \sim \mathsf{Bern}(p)$.

Example #2: Gaussian process

The second prominent example is the Gaussian process, in particular, being tailored for the i.i.d. case. In the i.i.d. case, it is named the i.i.d. Gaussian process. We say that $(X_1, X_2, \ldots, X_n, \ldots)$ is i.i.d. Gaussian if X_i's are i.i.d., and each follows the Gaussian distribution, say $X_i \sim \mathcal{N}(\mu, \sigma^2)$:

$$f_{X_i}(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \qquad \forall x \in \mathbb{R} \text{ and } \forall i = \{1, 2, \ldots\}.$$

As hinted earlier, one important application of the i.i.d. Gaussian process is communication. Noise signals in communication can be modeled as the i.i.d. Gaussian process. We will develop this modeling in Part III.

Example #3: Markov process

The last example is arguably the most famous and useful stationary process, named the *Markov process*. This is named after Andrey Markov, a Russian mathematician in the 1900s. See Fig. 3. In reality, the i.i.d. assumption does not often hold. One such concrete example is an English text. Suppose we read the first and second letters as “t” and “h”, respectively. Then, the next letter would be highly likely to be “e” because there are many “the” in an English text. From this, we see that letters are highly dependent with each other.

Figure 3: Andrey Markov is a Russian mathematician who made important achievements w.r.t. a random process that people later named the Markov process.

On the other hand, the *stationarity* assumption is often practically relevant. Think about the same English text example. Notice that the statistics of a 10-year-old text would be almost the same as that of a nowadays text; for instance, the frequency of “the” in an old text would be roughly the same as that of a current text. The Markov process is one such stationary process that well respects this scenario.

We say that $(X_1, X_2, \ldots, X_n, \ldots)$ is a Markov process if it satisfies the following certain condition:

$$\mathbb{P}(x_{m+1}|x_m, x_{m-1}, \ldots, x_1) = \mathbb{P}(x_{m+1}|x_m). \tag{4}$$

Here what this condition means is that given the current value x_m, the future x_{m+1} and the past values $(x_{m-1}, \ldots, x_1)$ are independent with each other. The key condition is called the *Markov* property. In fact, the dependency of random variables in reality can be much more complicated than that imposed by the simple Markov property (4). It turns out that we can properly capture such complicated yet more realistic dependency by invoking a so-called *generalized*

Markov process that is characterized by

$$\mathbb{P}(x_{m+1}|x_m,\ldots,x_{m-\ell+1},x_{m-\ell},\ldots,x_1)=\mathbb{P}(x_{m+1}|x_m,\ldots,x_{m-\ell+1}). \tag{5}$$

Observe that the dependency of x_{m+1} on the past is now through possibly more past values, say ℓ values, $x_m,\ldots,x_{m-\ell+1}$. One can readily see that this is indeed a generalized Markov process, as it subsumes the Markov process as a special case of $\ell=1$.

As I mentioned earlier, *voice signals* in speech recognition can be modeled as a generalized Markov process. We will discuss details on this in Part III.

Visualization of the Markov process

In statistics and machine learning, the Markov property (4) is usually represented by an insightful picture which illustrates the relation across random variables. That is, a *graphical model*. In fact, the graphical model is concerning a generic random process, say $(X_1,X_2,\ldots,X_n)$, not limited to the Markov process. It captures the statistical structure of a random process with two entities: (1) *nodes* (corresponding to random variables); (2) *edges* (reflecting the dependency of a pair of two random variables involved). One interpretation of a graph is as follows. If we can disconnect the interested graph into two subgraphs $\mathcal{G}_1$ and $\mathcal{G}_2$ by removing a certain node, say X_i, then we say that the random variables in $\mathcal{G}_1$ are independent of those in $\mathcal{G}_2$, *conditioned on* X_i. For instance, consider (X_1,X_2,X_3) with $\mathbb{P}(x_1,x_2,x_3)=\mathbb{P}(x_1)\mathbb{P}(x_2|x_1)\mathbb{P}(x_3|x_2)$. One can easily see from the relation that $\mathbb{P}(x_3|x_2,x_1)=\mathbb{P}(x_3|x_2)$, which in turn implies that

$$\begin{aligned}\mathbb{P}(x_1,x_3|x_2)&=\mathbb{P}(x_1|x_2)\mathbb{P}(x_3|x_2,x_1)\\&=\mathbb{P}(x_1|x_2)\mathbb{P}(x_3|x_2)\end{aligned} \tag{6}$$

where the 1st equality is due the definition of condition probability; and the 2nd equality follows from $\mathbb{P}(x_3|x_2,x_1)=\mathbb{P}(x_3|x_2)$. This then implies that X_1 are X_3 are independent conditioned on X_2. So the graph is illustrated as:

$$X_1 - X_2 - X_3. \tag{7}$$

Note that the removal of X_2 disconnects X_1 and X_3. Applying this logic to the Markov process $(X_1,X_2,\ldots,X_n,\ldots)$, we can then obtain the graphical model as:

$$X_1 - X_2 - X_3 - \cdots - X_n - \cdots \tag{8}$$

Observe that if we remove any x_m, then x_{m+1} and $(x_1,\ldots,x_{m-1})$ are disconnected, thus implying that x_{m+1} and $(x_1,\ldots,x_{m-1})$ are independent conditioned on x_m. This is called the *Markov chain* as it looks like a chain. Some curious students may wonder whether *no directionality* of the chain (8) implies that given the current value x_m, the one-step past value is independent of all the future values:

$$\mathbb{P}(x_{m-1}|x_m,x_{m+1},x_{m+2},\ldots)=\mathbb{P}(x_{m-1}|x_m). \tag{9}$$

It turns out this is the case. Please check this in PS4.

Look ahead

Actually, there are many other interesting examples and concepts w.r.t. random processes. But we will stop here because studying further details may distract you so that you may lose interest in this important topic. Just keep in mind that there are many things to learn about and it is crucial that you should be strong at random processes. In fact, you will have a great chance to know more via a graduate-level course: EE528 Engineering Random Processes. I strongly recommend you to take the course later on if you wish to do something related to probability.

Instead we will move forward towards the two main targets: MAP and ML estimation principles. Next time, we will investigate the MAP estimation.

Lecture 13: MAP principle

Recap

Last time, we investigated a bit tricky concept: random process. While it is a non-trivial animal, its definition is very simple to state. It is just a *sequence of random variables*. Among many kinds of random processes, we focused on the following two types: (i) stationary process wherein every shifted version has the same joint distribution; and (ii) i.i.d. process, a special yet often-appearing instance of the stationary process in which the associated random variables are mutually *independent* and *identically distributed*. I then put a particular emphasis on the following three examples that have something to do with the focused applications to be dealt with later on: (i) Bernoulli process; (ii) i.i.d. Gaussian process; and (iii) Markov process.

We are now ready to study the two key principles that I mentioned several times earlier: (i) MAP (Maximum A Posteriori) estimation; and (ii) ML (Maximum Likelihood) estimation.

Today's lecture

Today we will investigate the first: the *MAP principle*. This lecture consists of four parts. In fact, the *cancer testing problem* that we explored in Lecture 4 is a very good example which helps us to figure out the essence of the MAP principle. In the first part, we will simply recall the problem setup. By introducing proper notations, I will then re-interpret several concepts that we already saw in Lecture 4, yet via a different language that is usually employed in the context of the MAP principle. Next I will emphasize the last key concept that plays a central role in MAP: *A Posteriori probability*. Lastly we will study the MAP estimation.

Revisit: Cancer testing problem

Here is the setup of the cancer testing problem. See Fig. 1. A person is tested for cancer disease.

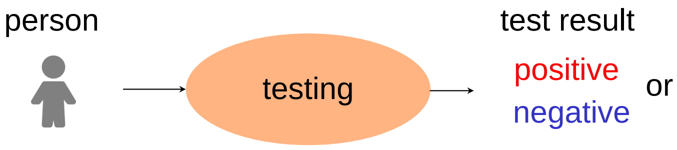


Figure 1: Cancer testing problem.

The test result reads either positive or negative. In Lecture 4, the key question that we were interested in was: What is the probability that the person has indeed cancer given that the test result is positive?

$$\mathbb{P}(\text{has indeed cancer}|\text{positive test}). \tag{1}$$

Now let us introduce some notations that will help us to relate the interested probability (1) to the MAP principle. Let X be a binary random variable that indicates whether a person has indeed cancer: $X = 1$ for cancer; 0 otherwise. Let Y be another binary random variable for a test result: $Y = 1$ for positive; $Y = 0$ for negative. See Fig. 2. Prior to digging into (1), let us

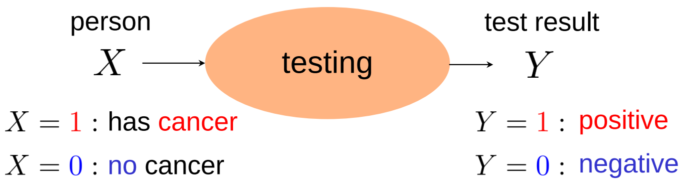


Figure 2: Re-interpretation of the center testing problem with new notations: X indicates whether a person has indeed cancer; Y denotes the test result.

first interpret several quantities that appeared in the course of computing the probability (1).

A priori probability

The first quantity is the *cancer population ratio*. In terms of X, it reads:

$$\mathbb{P}(X = 1) =: p. \tag{2}$$

In fact, this is one of the very important concepts. It is so called the *"a priori probability"*. Here "Priori" is a Latin word that means "before". So it can be interpreted as the probability we have *prior knowledge* on. In Lecture 4, we made a natural assumption that the probability is given as a small quantity like $p = 0.1$.

True Positive Rate (TPR) and False Negative Rate (FNR)

Remember we investigated two quantities that can be inferred from many clinical trials. One is the True Positive Rate (TPR), the probability that the test result is positive for a cancer individual. In terms of (X, Y) notations, it can be written as:

$$\mathbb{P}(Y = 1|X = 1). \tag{3}$$

The other is the False Negative Rate (FNR), or called the misdetection rate:

$$\mathbb{P}(Y = 0|X = 1). \tag{4}$$

Obviously we desire to have very high TPR, i.e., very small FNR. In the past example, we assumed that the misdetection rate is 5%, FNR is 0.05, which in turn leads to TPR $= 0.95$.

False Positive Rate (FPR) and True Negative Rate (TNR)

There were two more statistical quantities but now w.r.t. *normal* population. The first is the probability that the test reads positive for a normal person: the False Positive Rate (FPR), or called the false alarm rate:

$$\mathbb{P}(Y = 1|X = 0). \tag{5}$$

The other is the True Negative Rate (TNR): $\mathbb{P}(Y = 0|X = 0)$.

Tradeoff between FPR and TPR

Of course, it is good to have a small FPR (i.e., a large TNR). In reality, however, targeting too small FPR can be problematic. Why? To see this clearly, consider one extreme case where we aim at *exactly zero FPR*. In order to ensure zero FPR, the test result should be *always negative* no matter what and whatsoever. Otherwise it would be strictly positive as long as there is normal population. This then leads to zero TPR as well, which is definitely not a desired situation. Actually there is a *tradeoff* relationship between FPR and TPR:

$$\text{Tradeoff:} \quad \mathbb{P}(Y=1|X=0) \downarrow \implies \mathbb{P}(Y=1|X=1) \downarrow.$$

In order to decrease FPR, the test should be designed so that the event of declaring positive $(Y=1)$ is less likely. But this affects TPR negatively (i.e., decreases TPR). On the other hand, in order to increase TPR, the test should be designed to yield a *high chance of positive result*, which in turns affects FPR negatively (i.e., increases FPR).

Then, what can we do? Here one good thing about FPR is that FPR does not have to be very small because false alarm for cancer is acceptable in reality. False alarm is indeed annoying, but it is endurable because it has nothing to do with death-or-live matters. On the other hand, TPR *should be high enough*, as the misdetection is indeed disastrous to cancer person. Hence, it is crucial to find a good balanced point between the two. One typical rule-of-thumb is to increase FPR (sacrifice for a non-fatal measure) up to a point where TPR degradation is minimized. In the past example, the exemplary desired values for FPR and TPR were:

$$\begin{aligned} &\text{FPR: } \mathbb{P}(Y=1|X=0)=0.2 && \text{not very small, but somewhat small;} \\ &\text{TPR: } \mathbb{P}(Y=1|X=0)=0.95 && \text{very close to 1.} \end{aligned}$$

In general, the test should be designed so as to respect the following configuration:

$$\begin{aligned} \text{TPR}: &\quad \mathbb{P}(Y=1|X=1)=1-\epsilon_1 \\ \text{FNR}: &\quad \mathbb{P}(Y=0|X=1)=\epsilon_1 \qquad \text{very small} \\ \text{FPR}: &\quad \mathbb{P}(Y=1|X=0)=\epsilon_2 \qquad \text{somewhat small} \\ \text{TNR}: &\quad \mathbb{P}(Y=0|X=0)=1-\epsilon_2. \end{aligned} \tag{6}$$

A Posteriori probability

We are now ready to attack the targeted probability (1). In terms of (X,Y) notations, it is written as:

$$\mathbb{P}(X=1|Y=1). \tag{7}$$

Notice that it takes a similar expression, compared to the "a priori probability" $\mathbb{P}(X=1)$. The distinction is that it has a conditioned event $Y=1$ (marked in purple). Actually one can interpret the test result $Y=1$ as a sort of *observation*. Hence, the probability (7) can be interpreted as the one *after* making an observation. People wanted to use a similar Latin-style name for the probability (7). "A posteriori" is a Latin word that means "after". Hence, it is named the "A posteriori probability". This is the last key concept that plays a central role in the MAP principle.

Inference problem

The MAP principle arises in the context of a so-called *inference problem*. An *inference* problem is defined as the one wherein the goal is to *infer* an interested entity when the entity is *probabilistically related to* the observation given in the problem. In the context of cancer testing, a natural inference problem that we can think of is the one illustrated in Fig. 3. Here we wish to infer X (the ground truth indicting whether a person has cancer) from the observation Y (test result) which has a *statistical relationship* with X. Remember that $\mathbb{P}(Y=y|X=x)$'s in (6) (for $x,y \in \{0,1\}$) capture the statistical relationship, and the conditional probabilities (determined

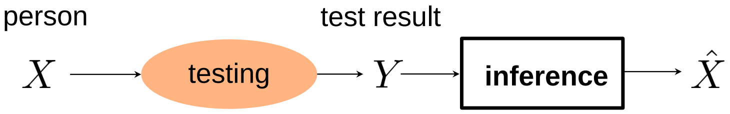


Figure 3: An inference problem.

by ϵ_1 and ϵ_2) are fixed once the test mechanism is designed. The "inference" block in Fig. 3 takes Y as an input to yield an estimate $\hat{X}$. Notice that $\hat{X}$ is a function of a random variable Y; hence, it is also a random variable. But once the test result is revealed as $Y = y$ ($y \in \{0, 1\}$), it becomes a fixed value, as long as there is no new randomness introduced in the inference block. Here we call such small y, a *realization* of the random variable Y.

MAP estimation

Suppose the test result is revealed as $Y = y$ where $y \in \{0, 1\}$. Given the realization $Y = y$, the optimal inference (estimation) for X is defined as the one that maximizes the *correction decision probability*:

$$\mathbb{P}(X = \hat{X} | Y = y).$$

Notice that $X = \hat{X}$ is an event that we make a correct decision. Since the observation $Y = y$ is *given* in the problem, we take into consideration the *conditioned version*. Here $\hat{X}$ is the one that we can choose as per our inference mechanism. In cancer testing, there are only two choices for X between 0 and 1. Hence, the optimal estimation for X can be written as:

$$\hat{X}_{\sf opt} := \arg \max_{\hat{x} \in \{0,1\}} \underbrace{\mathbb{P}(X = \hat{x} | Y = y)}_{\text{A Posteriori probability}} \tag{8}$$

where the notation "arg max" means "is the one that maximizes". Here the small $\hat{x}$ is a dummy variable that serves as a candidate that $\hat{X}$ can take on.

Note that the objective probability (colored in blue) is the "A posteriori probability". Hence, one can interpret the optimal estimator as the one that *Maximizes A Posteriori probability* (MAP). So, another name of the optimal estimator is the MAP estimator, $\hat{X}_{\sf MAP}$:

$$\hat{X}_{\sf opt} = \hat{X}_{\sf MAP} = \arg \max_{\hat{x} \in \{0,1\}} \mathbb{P}(X = \hat{x} | Y = y). \tag{9}$$

Derivation of the MAP estimation

How to compute $\hat{X}_{\sf MAP}$? First observe that the "a posteriori probability" can be written as:

$$\mathbb{P}(X = \hat{x} | Y = y) = \frac{\mathbb{P}(X = \hat{x}, Y = y)}{\mathbb{P}(Y = y)}.$$

This is due to the definition of conditional probability. Here one key observation that we can make is that the denominator $\mathbb{P}(Y = y)$ has nothing to do with the variation of $X = \hat{x}$, i.e., it is not a function $X = \hat{x}$. Hence, we can simplify the MAP estimator (9) as:

$$\begin{aligned} \hat{X}_{\sf MAP} &= \arg \max_{\hat{x} \in \{0,1\}} \mathbb{P}(X = \hat{x}, Y = y) \\ &= \arg \max_{\hat{x} \in \{0,1\}} \mathbb{P}(X = \hat{x}) \mathbb{P}(Y = y | X = \hat{x}) \end{aligned} \tag{10}$$

where the 2nd equality is due to the definition of conditional probability. Using (6), we can express the objective probability as:

$$\mathbb{P}(X=1)\mathbb{P}(Y=y|X=1) = \begin{cases} p(1-\epsilon_1), & \text{if } y=1; \\ p\epsilon_1, & \text{if } y=0, \end{cases}$$
$$\mathbb{P}(X=0)\mathbb{P}(Y=y|X=0) = \begin{cases} p\epsilon_2, & \text{if } y=1; \\ p(1-\epsilon_2), & \text{if } y=0. \end{cases}$$

A more succinct representation for the above is:

$$\begin{aligned} \mathbb{P}(X=1)\mathbb{P}(Y=y|X=1) &= p(1-\epsilon_1)^y {\epsilon_1}^{1-y}; \\ \mathbb{P}(X=0)\mathbb{P}(Y=y|X=0) &= (1-p)(1-\epsilon_2)^{1-y} {\epsilon_2}^y. \end{aligned} \tag{11}$$

Applying this to (10), we obtain the explicit MAP decision rule:

$$p(1-\epsilon_1)^y {\epsilon_1}^{1-y} \underset{\hat{X}_{\mathsf{MAP}}=0}{\overset{\hat{X}_{\mathsf{MAP}}=1}{\gtrless}} (1-p)(1-\epsilon_2)^{1-y} {\epsilon_2}^y. \tag{12}$$

Here you may wonder what if the LHS and RHS are exactly the same. In such a very rare case, we flip a fair coin to decide, or simply declare one (say $\hat{X}_{\mathsf{MAP}} = 1$) out of the two. Even such a dumb-looking decision does not violate the MAP decision rule (taking the one that maximizes a posteriori probability), as both the a posteriori probabilities are the same.

A different expression of the MAP decision rule (12)

Someone may want to obtain a different yet possibly insightful expression for (12). Taking an increasing function $\log(\cdot)$ on both sides in (12), we get:

$$\log p + y\log(1-\epsilon_1) + (1-y)\log\epsilon_1 \underset{\hat{X}_{\mathsf{MAP}}=0}{\overset{\hat{X}_{\mathsf{MAP}}=1}{\gtrless}} \log(1-p) + (1-y)\log(1-\epsilon_2) + y\log\epsilon_2. \tag{13}$$

Massaging the above a bit, we obtain:

$$y\log\frac{(1-\epsilon_1)(1-\epsilon_2)}{\epsilon_1\epsilon_2} \underset{\hat{X}_{\mathsf{MAP}}=0}{\overset{\hat{X}_{\mathsf{MAP}}=1}{\gtrless}} \log\frac{1-p}{p} + \log\frac{1-\epsilon_2}{\epsilon_1}. \tag{14}$$

Since the multiplied term $\log\frac{(1-\epsilon_1)(1-\epsilon_2)}{\epsilon_1\epsilon_2}$ in the LHS is positive in a natural test design setting where $\epsilon_1 < \frac{1}{2}$ and $\epsilon_2 < \frac{1}{2}$, we get:

$$y \underset{\hat{X}_{\mathsf{MAP}}=0}{\overset{\hat{X}_{\mathsf{MAP}}=1}{\gtrless}} \frac{\log\frac{1-p}{p}}{\log\frac{(1-\epsilon_1)(1-\epsilon_2)}{\epsilon_1\epsilon_2}} + \frac{\log\frac{1-\epsilon_2}{\epsilon_1}}{\log\frac{(1-\epsilon_1)(1-\epsilon_2)}{\epsilon_1\epsilon_2}}. \tag{15}$$

Example: $p = 0.1, \epsilon_1 = 0.05, \epsilon_2 = 0.2$

Let us see how the MAP estimation (15) works in the past example where $p = 0.1, \epsilon_1 = 0.05$, $\epsilon_2 = 0.2$. In this case,

$$\frac{\log\frac{1-p}{p}}{\log\frac{(1-\epsilon_1)(1-\epsilon_2)}{\epsilon_1\epsilon_2}} \approx 0.5074 \quad \frac{\log\frac{(1-\epsilon_2)}{\epsilon_1}}{\log\frac{(1-\epsilon_1)(1-\epsilon_2)}{\epsilon_1\epsilon_2}} \approx 0.6402.$$

Putting this into (15),

$$y \overset{1}{\underset{0}{\gtrless}} 0.5074 + 0.6402 = 1.1476. \tag{16}$$

In this case, we declare $\hat{X}_{\sf MAP} = 0$ no matter what the test result y is. This looks like a dumb decision, although it is indeed the optimal decision rule and we did not make any mistake in the derivation. Then, why does this happen? This is because the "a priori probability" $p = 0.1$ (cancer population) affects the decision significantly. In (16), the number 0.6402 (colored in green) is a quantity that depends solely on test accuracy, as it is a function of ϵ_1 and ϵ_2. On the other hand, the other number 0.5074 (colored in purple) is a quantity that is affected by a priori probability p. If there were no bias in cancer population ratio, i.e., $p = 0.5$, the purple quantity would be 0. In this case, we make a plausible decision: declaring 1 if positive; 0 otherwise. However, in the biased yet realistic situation $p = 0.1$, the number 0.5074 makes the decision threshold above 1, leading to the dumb-looking decision $\hat{X}_{\sf MAP} = 0$.

This decision definitely relieves the person who got positive, to some extent. But not fully because the decision looks silly: declaring 0 no matter what.

Look ahead

Now one natural question that arises in this context is then: Is there any other way that is more reliable and trustable? Next time, we will address this question.

EE210 Probability and Introductory Random Processes
KAIST, Spring 2021

April 15, 2021
Changho Suh (chsuh@kaist.ac.kr)

Lecture 14: MAP under multiple observations

Recap

Last time, we explored one of the two key principles: MAP estimation. We figured out the essence of the principle in the context of the inference problem for cancer prediction. See Fig. 1. Given a realization of test result $Y = y \in \{0, 1\}$, the optimal estimator is proven to be the MAP estimator that Maximizes A Posteriori probability:

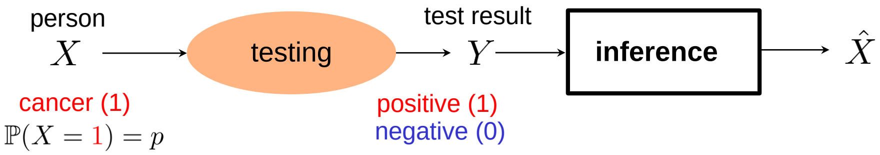


Figure 1: Inference problem for cancer prediction with a priori knowledge on cancer population ratio $\mathbb{P}(X = 1) = p$. The goal of the problem is to infer whether an interested person has cancer $X = 1$ given a test result Y.

$$\begin{aligned}\hat{X}_{\mathsf{opt}} = \hat{X}_{\mathsf{MAP}} &= \arg\max_{\hat{x}\in\{0,1\}} \underbrace{\mathbb{P}(X=\hat{x}|Y=y)}_{\text{A Posteriori probability}} \\ &= \arg\max_{\hat{x}\in\{0,1\}} \mathbb{P}(X=\hat{x})\mathbb{P}(Y=y|X=\hat{x})\end{aligned} \tag{1}$$

where the 2nd equality is because $\mathbb{P}(X = \hat{x}|Y = y) = \frac{\mathbb{P}(X=\hat{x},Y=y)}{\mathbb{P}(Y=y)}$ and the denominator $\mathbb{P}(Y = y)$ is not a function of $X = \hat{x}$. Under the following reasonable setting:

$$\begin{aligned}\text{TPR}: &\quad \mathbb{P}(Y=1|X=1) = 1-\epsilon_1 \\ \text{FNR}: &\quad \mathbb{P}(Y=0|X=1) = \epsilon_1 \qquad \text{very small} \\ \text{FPR}: &\quad \mathbb{P}(Y=1|X=0) = \epsilon_2 \qquad \text{somewhat small} \\ \text{TNR}: &\quad \mathbb{P}(Y=0|X=0) = 1-\epsilon_2,\end{aligned} \tag{2}$$

we then obtained the explicit MAP decision rule:

$$y \underset{\hat{X}_{\mathsf{MAP}}=0}{\overset{\hat{X}_{\mathsf{MAP}}=1}{\gtrless}} \frac{\log\frac{1-p}{p}}{\log\frac{(1-\epsilon_1)(1-\epsilon_2)}{\epsilon_1\epsilon_2}} + \frac{\log\frac{1-\epsilon_2}{\epsilon_1}}{\log\frac{(1-\epsilon_1)(1-\epsilon_2)}{\epsilon_1\epsilon_2}}. \tag{3}$$

At the end of the last lecture, however, I mentioned that this decision rule is sort of *untrustable* as it declares the same $\hat{X}_{\mathsf{MAP}} = 0$ no matter what and whatsoever, in realistic scenarios like the one where $p = 0.1$, $\epsilon_1 = 0.05$ and $\epsilon_2 = 0.2$:

$$y \underset{0}{\overset{1}{\gtrless}} 0.5074 + 0.6402 = 1.1476 \quad\longrightarrow\quad \hat{X}_{\mathsf{MAP}} = 0. \tag{4}$$

Even if the decision says "no cancer" under the positive test result, people would not believe this dumb-looking decision, as the estimator yields the same answer for *any* person tested and *any* test result. So a natural question is: Is there a trustable way to take?

Today's lecture

Today we will answer the question. It turns out there is an easy way to equip the test with trustability. That is, to *repeat* the test as many as possible. In this lecture, we will first derive the optimal decision rule (i.e., the MAP rule) under the multiple-test setting and then demonstrate that the derived rule gets indeed more trustable with an increase in the number of tests. Specifically what we are going to do are three folded. First of all, we will make a specific yet reasonable assumption in the multiple-test setting which turns out to ease the MAP derivation. Next we will derive the MAP decision rule. Lastly we will figure out the decision rule is indeed more reliable as the number of tests increases.

Multiple tests

Here is the multiple-test setting. See Fig. 2. Given the multiple observations of the test results, say $(Y_1, \ldots, Y_n) = (y_1, \ldots, y_n)$ where $y_i \in \{0, 1\}$ for all i's, we wish to infer X. As shown earlier, the optimal estimator is always MAP by the definition of the optimal estimation (the one that maximizes the correct decision probability, equivalently the a posteriori probability). Hence, we focus on the MAP estimator as depicted in Fig. 2.

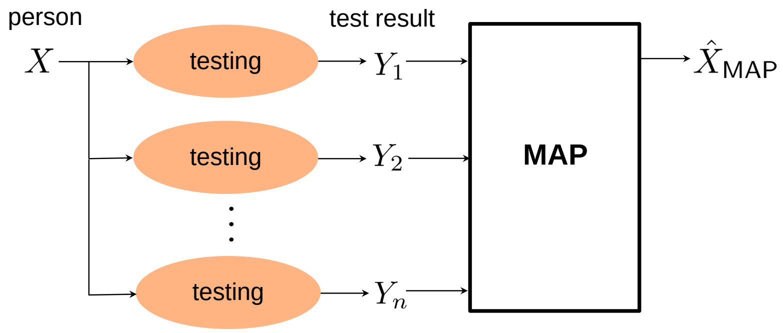


Figure 2: Inference problem for cancer prediction under multiple test results $(Y_1, \ldots, Y_n)$.

As mentioned earlier, we will make one important assumption. The assumption is based on the following observation. For the same person, test results depend solely on *testing environments* such as testing device variation per trial. Naturally one may imagine that such environments are *independent* across trials and each trial has the *same statistical behaviour*. Here we will take this as an assumption. Formally, we assume that Y_i's are *conditionally independent* and identically distributed given $X = x$:

$$\text{(Assumption): Given } X = x, \{Y_i\}_{i=1}^n \text{ are } i.i.d. \tag{5}$$

Recalling from (2) that $\mathbb{P}(Y_i = 1 | X = 1) = 1 - \epsilon_1$ and $\mathbb{P}(Y_i = 1 | X = 0) = \epsilon_2$, the assumption implies:

$$\begin{aligned} Y_i &\sim \mathsf{Bern}(1 - \epsilon_1) \qquad && \text{when } X = 1; \\ Y_i &\sim \mathsf{Bern}(\epsilon_2) && \text{when } X = 0. \end{aligned} \tag{6}$$

MAP derivation

Given $(Y_1, \ldots, Y_n) = (y_1, \ldots, y_n)$, the MAP estimator reads:

$$\hat{X}_{\text{MAP}} = \arg \max_{\hat{x} \in \{0,1\}} \mathbb{P}(X = \hat{x} | Y_1 = y_1, \ldots, Y_n = y_n). \tag{7}$$

The only distinction w.r.t. the single-test case is that now the conditioned event includes many test results. Again, due to the definition of conditional probability,

$$\mathbb{P}(X=\hat{x}|Y_1=y_1,\dots,Y_n=y_n)=\frac{\mathbb{P}(X=\hat{x},Y_1=y_1,\dots,Y_n=y_n)}{\mathbb{P}(Y_1=y_1,\dots,Y_n=y_n)}.$$

Since $\mathbb{P}(Y_1=y_1,\dots,Y_n=y_n)$ is not a function of $X=\hat{x}$, the estimator (7) can be written as:

$$\hat{X}_{\mathsf{MAP}}=\arg\max_{\hat{x}\in\{0,1\}}\mathbb{P}(X=\hat{x})\mathbb{P}(Y_1=y_1,\dots,Y_n=y_n|X=\hat{x}). \tag{8}$$

Due to the i.i.d. assumption (5) of $\{Y_i\}_{i=1}^n$ given $X=\hat{x}$, $\mathbb{P}(Y_1=y_1,\dots,Y_n=y_n|X=\hat{x})=\prod_{i=1}^n\mathbb{P}(Y_i=y_i|X=\hat{x})$. This then yields:

$$\hat{X}_{\mathsf{MAP}}=\arg\max_{\hat{x}\in\{0,1\}}\mathbb{P}(X=\hat{x})\prod_{i=1}^n\mathbb{P}(Y_i=y_i|X=\hat{x}). \tag{9}$$

Using (2), we can then compute the objective probability in the above as:

$$\begin{aligned}\mathbb{P}(X=1)\prod_{i=1}^n\mathbb{P}(Y_i=y_i|X=1)&=p(1-\epsilon_1)^{\sum_{i=1}^n y_i}\epsilon_1{}^{n-\sum_{i=1}^n y_i};\\ \mathbb{P}(X=0)\prod_{i=1}^n\mathbb{P}(Y_i=y_i|X=0)&=(1-p)(1-\epsilon_2)^{n-\sum_{i=1}^n y_i}\epsilon_2{}^{\sum_{i=1}^n y_i}.\end{aligned} \tag{10}$$

This expression is very similar to the single-test-case counterpart; see (11) in Lecture 13. The key distinction lies in the exponents in the RHSs. Instead of y, we have $\sum_{i=1}^n y_i$. For notational simplicity, let $s:=\sum_{i=1}^n y_i$. Then, the MAP rule can be simplified as:

$$p(1-\epsilon_1)^s\epsilon_1{}^{n-s}\underset{\hat{X}_{\mathsf{MAP}}=0}{\overset{\hat{X}_{\mathsf{MAP}}=1}{\gtrless}}(1-p)(1-\epsilon_2)^{n-s}\epsilon_2{}^s. \tag{11}$$

Like the single-test case, we can massage this to obtain a more insightful expression. Taking $\log(\cdot)$ on both sides, we get:

$$\log p+s\log(1-\epsilon_1)+(n-s)\log\epsilon_1\underset{0}{\overset{1}{\gtrless}}\log(1-p)+(n-s)\log(1-\epsilon_2)+s\log\epsilon_2. \tag{12}$$

Aggregating the terms w.r.t. s in the LHS, we obtain:

$$s\log\frac{(1-\epsilon_1)(1-\epsilon_2)}{\epsilon_1\epsilon_2}\underset{0}{\overset{1}{\gtrless}}n\log\frac{1-\epsilon_2}{\epsilon_1}+\log\frac{1-p}{p}. \tag{13}$$

Since $\log\frac{(1-\epsilon_1)(1-\epsilon_2)}{\epsilon_1\epsilon_2}$ is positive in an interested setting ($\epsilon_1<\frac{1}{2}$ and $\epsilon_2<\frac{1}{2}$), we get:

$$\underbrace{\frac{s}{n}}_{\text{fraction of postive tests}}\underset{0}{\overset{1}{\gtrless}}\underbrace{\frac{\log\frac{1-\epsilon_2}{\epsilon_1}}{\log\frac{(1-\epsilon_1)(1-\epsilon_2)}{\epsilon_1\epsilon_2}}+\frac{1}{n}\cdot\frac{\log\frac{1-p}{p}}{\log\frac{(1-\epsilon_1)(1-\epsilon_2)}{\epsilon_1\epsilon_2}}}_{\text{threshold}}. \tag{14}$$

Here we intentionally divide both by n, in order to read the LHS as an intuitive term: the *empirical average* $\frac{s}{n}=\frac{\sum_{i=1}^n y_i}{n}$ (the fraction of positive test results). This way, we see that this decision rule makes an intuitive sense. If the fraction of positive results is *above* the threshold (RHS), we declare $\hat{X}_{\mathsf{MAP}} = 1$; otherwise $\hat{X}_{\mathsf{MAP}} = 0$.

Reliability of the MAP estimator (14)

In order to have a deeper understanding on the threshold (the RHS in (14)) in light of the reliability of the MAP estimator, let's focus on the past example: $p = 0.1$, $\epsilon_1 = 0.05$ and $\epsilon_2 = 0.2$. In this case, the two quantities in the RHS are around:

$$\frac{\log \frac{1-\epsilon_2}{\epsilon_1}}{\log \frac{(1-\epsilon_1)(1-\epsilon_2)}{\epsilon_1 \epsilon_2}} \approx 0.6402. \quad \frac{\log \frac{1-p}{p}}{\log \frac{(1-\epsilon_1)(1-\epsilon_2)}{\epsilon_1 \epsilon_2}} \approx 0.5074.$$

Putting this into (14), the rule can be simply stated as:

$$\frac{s}{n} \overset{1}{\underset{0}{\gtrless}} 0.6402 + \frac{0.5074}{n}. \tag{15}$$

The number 0.6402 (colored in green) is a quantity that depends solely on *test accuracy*, as it is a function of ϵ_1 and ϵ_2. On the other hand, the other number 0.5074 (colored in purple) is a quantity that is also affected by the *a priori probability* p.

In the single-test case $n = 1$, the RHS in (15) is $0.6402 + \frac{0.5074}{1} = 1.1476$ (exceeding 1); therefore the MAP rule declares $\hat{X}_{\mathsf{MAP}} = 0$ all the time. In the multiple-test case, however, as n increases, the term $\frac{0.5074}{n}$ (depending on the a priori probability p) vanishes; hence, the MAP rule becomes dominant solely by the first term 0.6402 that captures test accuracy dictated by ϵ_1 and ϵ_2. If the test accuracies w.r.t. cancer and normal populations were the same, i.e., $\epsilon_1 = \epsilon_2$, then the green term would be 0.5:

$$\frac{\log \frac{1-\epsilon_2}{\epsilon_1}}{\log \frac{(1-\epsilon_1)(1-\epsilon_2)}{\epsilon_1 \epsilon_2}} = \frac{\log \frac{1-\epsilon_1}{\epsilon_1}}{2 \log \frac{1-\epsilon_1}{\epsilon_1}} = \frac{1}{2}, \tag{16}$$

which is exactly "majority voting".

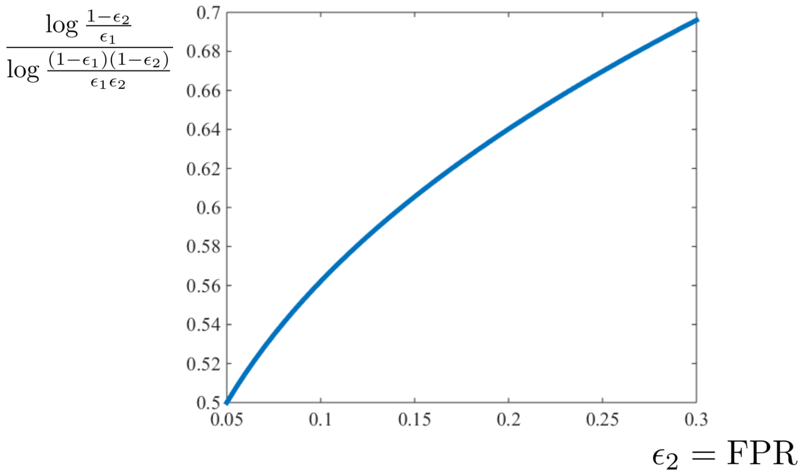


Figure 3: Decision threshold as a function of ϵ_2 (False Positive Rate) when $1 - \text{TPR} = \epsilon_1 = 0.05$ and $n \to \infty$.

Since the test is designed to allow for a bit larger ϵ_2 (relative to ϵ_1) due to the tradeoff relationship between TPR and FPR, the tests would yield positive outcomes more likely; hence, the threshold

0.6402 here is set to be larger than 0.5. For a larger ϵ_2, the threshold would be higher; see Fig. 3. Here we plot the green term as a function of ϵ_2 when $\epsilon_1 = 0.05$. Notice that when $\epsilon_2 = \epsilon_1 = 0.05$, the threshold is exactly 0.5 and it grows with an increase in ϵ_2. In the focused example $\epsilon_2 = 0.2$, the MAP estimator provide a sort of "a bit biased majority voting" in the limit of n: declaring $\hat{X}_{\sf MAP} = 1$ with more than $\approx 64\%$ positive test results.

From the above example, we see that the MAP rule depends only on *test accuracy* with an increase in n; hence, it would be more trustable as n grows, as long as the test accuracy is better than that of random guessing, i.e., $\epsilon_1 < \frac{1}{2}, \epsilon_2 < \frac{1}{2}$.

Look ahead

During the past two lectures, we have studied the MAP estimation via a certain yet prominent example: the inference problem for cancer prediction. As mentioned before, there are many instances where the MAP principle is quite instrumental. We will delve into a couple of such instances later in Part III. Instead next time, we will investigate the 2nd key principle: Maximum Likelihood (ML) estimation.

Changho Suh (chsuh@kaist.ac.kr)

Lecture 15: Maximum Likelihood Estimation (MLE)

Recap

During the past two lectures, we have explored the MAP principle in the context of the inference problem illustrated as in Fig. 1. In this setting, the essential fact that we figured out is that

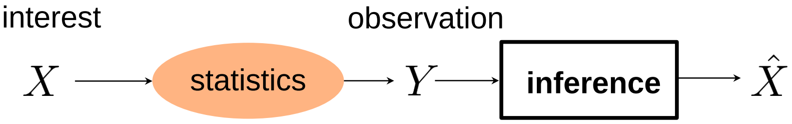


Figure 1: The goal of the inference problem is to infer whether an interested entity, usually denoted by X, from some observation Y that has a statistical relationship with X.

the optimal inference (defined as the one that maximizes the correction decision probability) is equivalent to the MAP estimation that Maximizes A Posteriori probability:

$$\begin{aligned}\hat{X}_{\sf opt} = \hat{X}_{\sf MAP} &= \arg\max_{\hat{x}\in\mathcal{X}} \underbrace{\mathbb{P}(X=\hat{x}|Y=y)}_{\text{A Posteriori probability}} \\ &= \arg\max_{\hat{x}\in\mathcal{X}} \mathbb{P}(X=\hat{x})\mathbb{P}(Y=y|X=\hat{x})\end{aligned} \tag{1}$$

where the 2nd equality is because $\mathbb{P}(X=\hat{x}|Y=y) = \frac{\mathbb{P}(X=\hat{x},Y=y)}{\mathbb{P}(Y=y)}$ and the denominator $\mathbb{P}(Y=y)$ is not a function of $X=\hat{x}$.

Here the key assumption that we made is:

$$\text{(Assumption): The “a priori probability” } \mathbb{P}(X=\hat{x}) \text{ is known!} \tag{2}$$

In reality, however, there are many inference scenarios in which we may not have such prior knowledge on the interested entity. This is where the Maximum Likelihood (ML) principle kicks in.

Today's lecture

Today we will investigate the ML principle in depth. In particular, we will cover the following four contents. We will first introduce one prominent setting wherein we have no prior knowledge on the statistics of an interested entity. It turns out the optimal inference in the no-prior-knowledge setting reduces the ML estimation. So in the second part, we will figure this out. Next we will derive the ML estimator. Lastly we will demonstrate that the ML estimator offers a reasonable good performance as the number of observations grows.

A parameter estimation setting

The prominent setting that we will focus on is *parameter estimation.* Let me explain what parameter estimation is in a very simple context that concerns the Bernoulli process with parameter p. See Fig. 2. As per the Bernoulli parameter p, i.i.d. samples $\{Y_i\}_{i=1}^n$ are generated.

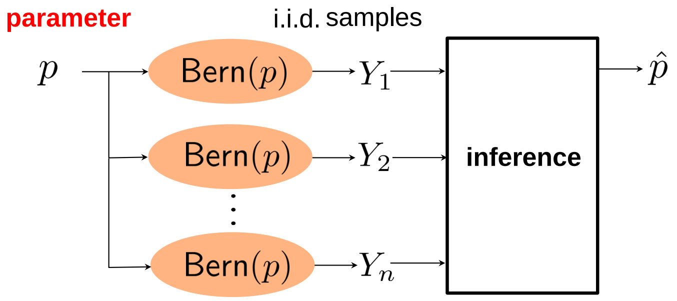


Figure 2: Bernoulli parameter p estimation from multiple i.i.d samples generated according to Bern(p).

The goal of the problem is to estimate the parameter p from such observations. In this problem setting, one key distinction w.r.t. the inference problem for cancer prediction with the prior knowledge of cancer population ratio is that usually we have no idea about the statistics of the parameter p. Since p is a *continuous* value, the distinction means that the *pdf* $f_p(\cdot)$ of p is unknown. Now what is the optimal inference for p in this setting? To figure this out, let us start with pondering "correction decision probability" which plays a key role in deriving the optimal estimate.

Correct-decision probability density

Consider the correct-decision probability: $\mathbb{P}(p = \hat{p})$. You can immediately see an issue here. The issue is that since p is continuous, the probability always reads 0:

$$\mathbb{P}(p = \hat{p}) = 0. \tag{3}$$

In order to have a meaningful non-zero quantity, we should consider its *density* as we did for continuous random variable:

$$\lim_{\delta \to 0} \frac{\mathbb{P}(p \in [\hat{p}, \hat{p} + \delta])}{\delta} =: f_p(\hat{p}). \tag{4}$$

Since the observations are given in the problem, we consider its *conditional* counterpart:

$$\lim_{\delta \to 0} \frac{\mathbb{P}(p \in [\hat{p}, \hat{p} + \delta])|Y_1 = y_1, \dots, Y_n = y_n)}{\delta} =: f_p(\hat{p}|Y_1 = y_1, \dots, Y_n = y_n). \tag{5}$$

Optimal estimator

With the conditional correct-decision probability density (5), we can define the optimal estimator as:

$$\hat{p}_{\mathsf{opt}} := \arg \max_{t \in [0,1]} f_p(t|Y_1 = y_1, \dots, Y_n = y_n) \tag{6}$$

where t indicates a dummy variable for the choice of $\hat{p}$ and $t \in [0,1]$ as it is a candidate for the Bernoulli parameter. Using the Bayes' law (applying the definition of conditional probability twice), we get:

$$\begin{aligned} f_p(t|Y_1 = y_1, \dots, Y_n = y_n) &= \frac{f_{p,Y_1,\dots,Y_n}(t, y_1, \dots, y_n)}{\mathbb{P}(Y_1 = y_1, \dots, Y_n = y_n)} \\ &= \frac{f_p(t)\mathbb{P}(Y_1 = y_1, \dots, Y_n = y_n|p = t)}{\mathbb{P}(Y_1 = y_1, \dots, Y_n = y_n)} \end{aligned} \tag{7}$$

where $f_{p,Y_1,\ldots,Y_n}(t,y_1,\ldots,y_n)$ denotes the joint distribution w.r.t. p and $\{Y_i\}_{i=1}^n$ defined as:

$$f_{p,Y_1,\ldots,Y_n}(t,y_1,\ldots,y_n) := \lim_{\delta\to 0}\frac{\mathbb{P}(p\in[t,t+\delta],Y_1=y_1,\ldots,Y_n=y_n)}{\delta}. \tag{8}$$

In (7), the denominator $\mathbb{P}(Y_1=y_1,\ldots,Y_n=y_n)$ is not a function of t. So the optimal estimator reduces to:

$$\hat{p}_{\sf opt} = \arg\max_{t\in[0,1]} f_p(t)\mathbb{P}(Y_1=y_1,\ldots,Y_n=y_n|p=t). \tag{9}$$

Now remember that we know nothing about the pdf $f_p(t)$. What can we do then? One reasonable way to take in such a case is to be *equally open for all possible choices*. In an effort to put no biased weight for a particular value, people often assume:

$$f_p(t) \text{ is uniformly distributed.} \tag{10}$$

Maximum Likelihood Estimation (MLE)

Since $f_p(t)$ is irrelevant to t under the assumption (10), the optimal estimator (9) reads:

$$\hat{p}_{\sf opt} = \arg\max_{t\in[0,1]} \underbrace{\mathbb{P}(Y_1=y_1,\ldots,Y_n=y_n|p=t)}_{\text{Likelihood}}. \tag{11}$$

Here the objective conditional probability (marked in blue)) is a very famous notion, called the *likelihood*. So one can interpret the estimator as the one that Maximizes Likelihood. Hence, it is called the *ML estimator*.

$$\hat{p}_{\sf opt} = \hat{p}_{\sf ML} = \arg\max_{t\in[0,1]} \mathbb{P}(Y_1=y_1,\ldots,Y_n=y_n|p=t). \tag{12}$$

MLE derivation

Now how to compute the MLE (12)? First observe that given $p=t$, $\{Y_i\}_{i=1}^n$ is i.i.d., each being according to $\mathsf{Bern}(t)$. Hence,

$$\begin{aligned}\mathbb{P}(Y_1=y_1,\ldots,Y_n=y_n|p=t) &= \prod_{i=1}^n \mathbb{P}(Y_i=y_i|p=t)\\ &= t^{(\text{\# of 1's})}(1-t)^{(\text{\# of 0's})}\\ &= t^{\sum_{i=1}^n y_i}(1-t)^{n-\sum_{i=1}^n y_i}.\end{aligned} \tag{13}$$

Let $s:=\sum_{i=1}^n y_i$. Using this and applying (13) to (12), the ML estimator can be simplified as:

$$\hat{p}_{\sf ML} = \arg\max_{t\in[0,1]} t^s(1-t)^{n-s}. \tag{14}$$

How to figure out the maximizer, say t^*, in the above? Notice that the objective function is non-negative and achieves 0 at $t=0$ and $t=1$. So one can readily image that it goes up and down as t moves from 0 to 1. It turns out it is indeed the case as illustrated in Fig. 3. The function in Fig. 3 is an example curve when $(n,s)=(4,2)$. Some very sharp and curious students might ask how we can guarantee there is only one "up-&-down" movement, not multiple "up-&-down"s. To answer this, we should check if there is only one *stationary point* in the open interval

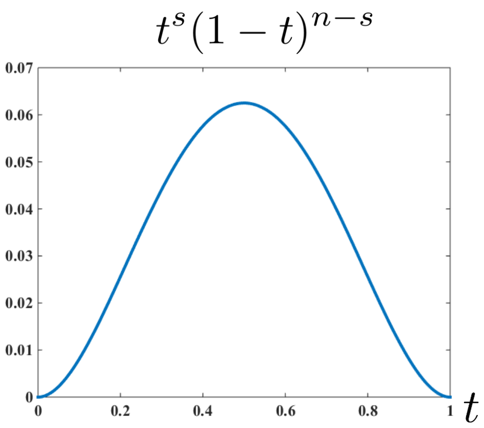


Figure 3: A shape of the objective $t^s(1-t)^{n-s}$ as a function of $t \in [0,1]$ when $(n,s)=(4,2)$.

$t \in (0,1)$. Here the *stationary point* is a very well-known terminology in the *optimization* field that indicates the point where the slope (derivative) is 0. So we should compute its derivative:

$$\begin{aligned}\frac{d}{dt}t^s(1-t)^{n-s} &= st^{s-1}(1-t)^{n-s} + t^s(n-s)(1-t)^{n-s-1}\cdot(-1)\\ &= t^{s-1}(1-t)^{n-s-1}\left\{s(1-t)-t(n-s)\right\}.\end{aligned} \tag{15}$$

where the 1st equality is due to the well-known property that you are familiar with: $(f\cdot g)' = f'\cdot g + f\cdot g'$. Observe that except for $t=0$ and $t=1$, there is only one stationary point t^* that satisfies:

$$s(1-t^*) - t^*(n-s) = 0. \tag{16}$$

Hence, the shape indeed looks like the one in Fig. 3. As you learned from the course on Calculus, its maximum point occurs at the stationary point:

$$t^* = \frac{s}{n} = \frac{\sum_{i=1}^n y_i}{n}. \tag{17}$$

Hence, the ML estimator is exactly the above. This is only for a particular realization $\{y_i\}_{i=1}^n$. So for the random process $\{Y_i\}_{i=1}^n$, the ML estimator should read:

$$\hat{p}_{\sf ML} = \frac{s}{n} = \frac{\sum_{i=1}^n Y_i}{n}. \tag{18}$$

If you think about it, the solution makes quite an intuitive sense. It coincides with the *sample mean* that you might guess initially as the optimal estimate.

Law of Large Numbers (LLN)

Now one natural question that you can ask is: Does $\hat{p}_{\sf ML}$ converge to the ground truth p as n goes to infinity? It turns out it is indeed the case. The ground for this is based on one very important law in the probability history, called the *Law of Large Numbers* (LLN for short). Here what the LLN says. Suppose $\{X_i\}_{i=1}^n$ is i.i.d. with $\mathbb{E}[X_i]=\mu$ and a *finite* $\mathsf{Var}(X_i)$ (often denoted by $\mathsf{Var}(X_i) < \infty$). Then, the sample mean of $\{X_i\}_{i=1}^n$ converges to its true mean μ *in probability* as n tends to infinity:

$$S_n := \frac{\sum_{i=1}^n X_i}{n} \xrightarrow{\text{in probability}} \mathbb{E}[X_i] = \mu. \tag{19}$$

Here we should be careful about stating the "convergence in probability". Notice that S_n is another *random variable*, as it is a function of random variables $\{X_i\}_{i=1}^n$. Hence the converged quantity is not guaranteed to be deterministic. So in order to compare potentially the non-deterministic converged quantity with the deterministic μ, we should make a *probabilistic statement*. Here what it means by "convergence in probability" is that

$$\mathbb{P}(|S_n - \mu| \geq \epsilon) \longrightarrow 0 \qquad \text{for any } \epsilon > 0. \tag{20}$$

Applying the LLN (19) to the Bernoulli process case of our interest, we obtain:

$$\hat{p}_{\mathsf{ML}} = \frac{\sum_{i=1}^n Y_i}{n} \xrightarrow{\text{in prob}} p. \tag{21}$$

Proof of LLN (19)

The proof of the LLN is straightforward if we rely upon one inequality technique that we learned: Chebyshev's inequality. Using Chebyshev's inequality, we get:

$$\begin{aligned}\mathbb{P}(|S_n - \mu| \geq \epsilon) &= \mathbb{P}(|S_n - \mathbb{E}[S_n]| \geq \epsilon) \\ &\leq \frac{\mathsf{Var}(S_n)}{\epsilon^2}\end{aligned} \tag{22}$$

where the 1st equality is due to $\mathbb{E}[S_n] = \frac{\mathbb{E}[X_1+\cdots+X_n]}{n} = \frac{n\mu}{\mu} = \mu$. Notice that

$$\begin{aligned}\mathsf{Var}(S_n) &= \mathsf{Var}\left(\frac{1}{n}\sum_{i=1} X_i\right) \\ &= \mathbb{E}\left[\frac{1}{n^2}\left(\sum_{i=1} X_i\right)^2\right] - \left(\mathbb{E}\left[\frac{1}{n}\sum_{i=1} X_i\right]\right)^2 \\ &= \frac{1}{n^2}\left(\mathbb{E}\left[\left(\sum_{i=1} X_i\right)^2\right] - \left(\mathbb{E}\left[\sum_{i=1} X_i\right]\right)^2\right) \\ &= \frac{1}{n^2}\sum_{i=1}^n \mathsf{Var}(X_i) \\ &= \frac{\mathsf{Var}(X_i)}{n}\end{aligned}$$

where the 2nd and 4th equalities are due to the useful fact for the variance; the 2nd last inequality comes from the independence of X_i's; and the last is because X_i's are identically distributed. Putting this to (22),

$$\begin{aligned}\mathbb{P}(|S_n - \mu| \geq \epsilon) &\leq \frac{\mathsf{Var}(S_n)}{\epsilon^2} \\ &= \frac{\mathsf{Var}(X_i)}{\epsilon^2 n} \longrightarrow 0 \qquad \text{as } n \to \infty.\end{aligned}$$

Look ahead

In fact, parameter estimation is a very important problem that arises in a wide variety of scenarios. One crucial problem that often appears in many contexts is: *Gaussian distribution* estimation. So next time, we will investigate the MLE for Gaussian distribution.

EE210 Probability and Introductory Random Processes April 29, 2021
KAIST, Spring 2021 Changho Suh (chsuh@kaist.ac.kr)

Lecture 16: MLE for Gaussian distribution

Recap

Last time we have studied the ML principle in the context of an important *parameter estimation* problem for the Bernoulli process, as illustrated in Fig. 1. According to the Bernoulli parameter $p \in [0,1]$, the i.i.d. samples $\{Y_i\}_{i=1}^n$ are generated. Given such generated samples, we wish to estimate the parameter p, assuming no prior knowledge on the statistics of p. We showed that the optimal estimate under the setting reduces to the ML estimate:

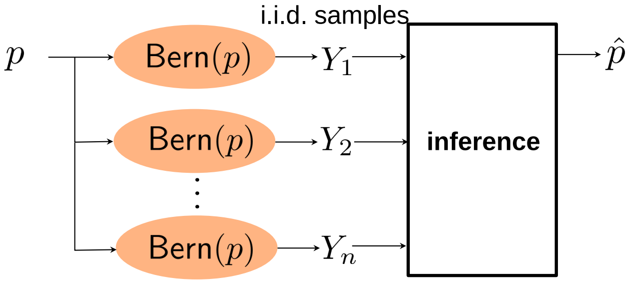


Figure 1: A Bernoulli parameter estimation problem.

$$\begin{aligned}\hat{p}_{\sf opt} = \hat{p}_{\sf ML} &= \arg\max_{t\in[0,1]} \underbrace{\mathbb{P}(Y_1 = y_1, \ldots, Y_n = y_n | p = t)}_{\text{likelihood}} \\ &= \arg\max_{t\in[0,1]} t^{\sum_{i=1}^n y_i}(1-t)^{n-\sum_{i=1}^n y_i}\end{aligned} \tag{1}$$

where the 2nd equality follows from the fact that $\{Y_i\}_{i=1}^n$ is i.i.d. $\sim \mathsf{Bern}(t)$ given $p = t$. Finding the *stationary* point in the open interval $(0,1)$ by taking the derivative w.r.t. t, we showed that the maximizer matches with our intuition, *sample mean*:

$$\hat{p}_{\sf ML} = \frac{\sum_{i=1}^n Y_i}{n}.$$

Using the Law of Large Numbers (LLN), we also proved that the ML estimate converges to the ground truth p (in probability) as the number n of observations tends to infinity:

$$\begin{aligned}&\frac{\sum_{i=1}^n Y_i}{n} \xrightarrow{\text{in prob}} \mathbb{E}[Y_i] = p \qquad \text{as } n \to \infty, \\ &\text{i.e., } \mathbb{P}\left(\left|\frac{\sum_{i=1}^n Y_i}{n} - p\right| \geq \epsilon\right) \xrightarrow{\text{as } n\to\infty} 0 \qquad \text{for any } \epsilon > 0.\end{aligned}$$

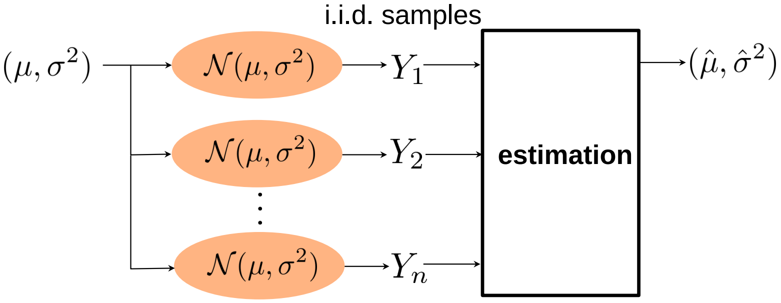
At the end of the last lecture, I emphasized that parameter estimation is a very important problem that often arises in statistics and machine learning. In particular, parameter estimation for *Gaussian distribution* often arises in practice, and therefore be of significant interest in a wide variety of contexts. This forms the content of today's lecture.

Today's lecture

Today we will explore the optimal estimation for Gaussian distribution parameters: mean and variance. Specifically what we are going to do are four folded. First off, we will introduce the problem setting and make some reasonable assumptions which turn out to ease the derivation of the optimal estimator. We will then show that the optimal estimator reduces to the ML estimator, as in the Bernoulli case. Next we will derive the ML estimator. Lastly we will demonstrate that the ML estimate converges to the ground truth as the number of observations grows.

Gaussian distribution estimation

Consider the problem setting for Gaussian parameter estimation, illustrated in Fig. 2. Let (μ, σ^2) be the mean and variance of the Gaussian distribution. The i.i.d. samples $\{Y_i\}_{i=1}^n$ are generated according to the parameters and are fed as input to the estimation block. As in the Bernoulli case, we assume no prior knowledge on the statistics of (μ, σ^2). As before, let us just start by computing "correct-decision probability" and figure out what to do without such prior knowledge.



i.i.d. samples
(μ, σ^2)
$\mathcal{N}(\mu, \sigma^2)$
Y_1
$\mathcal{N}(\mu, \sigma^2)$
Y_2
$\mathcal{N}(\mu, \sigma^2)$
Y_n
estimation
$(\hat{\mu}, \hat{\sigma}^2)$

Figure 2: Gaussian parameter estimation.

Correct-decision probability density

Again, since the interested entities are *continuous* values, the correct-decision probabilities are always zero:

$$\mathbb{P}(\mu = \hat{\mu}, \sigma = \hat{\sigma}) = 0. \tag{2}$$

So we should consider its *density* counterpart:

$$\lim_{\delta_1, \delta_2 \to 0} \frac{\mathbb{P}(\mu \in [\hat{\mu}, \hat{\mu} + \delta_1], \sigma \in [\hat{\sigma}, \hat{\sigma} + \delta_2])}{\delta_1 \delta_2} =: f_{\mu,\sigma}(\hat{\mu}, \hat{\sigma}). \tag{3}$$

Why divide by $\delta_1 \delta_2$ in the LHS? Remember how the pdf is defined:

$$\mathbb{P}(\mu \in [\hat{\mu}, \hat{\mu} + \delta_1], \sigma \in [\hat{\sigma}, \hat{\sigma} + \delta_2]) = \int_{\hat{\mu}}^{\hat{\mu}+\delta_1} \int_{\hat{\sigma}}^{\hat{\sigma}+\delta_2} f_{\mu,\sigma}(a, b) da db. \tag{4}$$

Notice that for very small values of δ_1 and δ_2, $\mathbb{P}(\mu \in [\hat{\mu}, \hat{\mu} + \delta_1], \sigma \in [\hat{\sigma}, \hat{\sigma} + \delta_2]) \approx f_{\mu,\sigma}(\hat{\mu}, \hat{\sigma})\delta_1 \delta_2$. So the pdf is defined as above in (3). Its conditioned counterpart reads:

$$f_{\mu,\sigma}(\hat{\mu}, \hat{\sigma} | Y_1 \in [y_1, y_1 + \delta_1], \ldots, Y_n \in [y_n, y_n + \delta_n]). \tag{5}$$

Here we take the *interval* (marked in purple) for the event w.r.t. Y_i's because $\{Y_i\}_{i=1}^n$ is a *continuous* random process. For illustrative purpose, we assume the same-size interval, say $\delta = \delta_1 = \cdots = \delta_n$. This assumption is okay, as we will drive it to the limit $\delta \to 0$ later on.

Optimal estimator

Using (5), we can then define the optimal estimator as:

$$(\hat{\mu}_{\mathsf{opt}}, \hat{\sigma}_{\mathsf{opt}}) := \arg\max_{a,b\in\mathbb{R}} f_{\mu,\sigma}(a, b|Y_1 \in [y_1, y_1+\delta], \dots, Y_n \in [y_n, y_n+\delta]). \tag{6}$$

where (a, b), marked in green, are dummy variables for $(\hat{\mu}, \hat{\sigma})$. Let δ be a very small value. Then, using the definition of conditional probability,

$$\begin{aligned} & f_{\mu,\sigma}(a, b|Y_1 \in [y_1, y_1+\delta], \dots, Y_n \in [y_n, y_n+\delta]) \\ &= \frac{f_{\mu,\sigma,Y_1,\dots,Y_n}(a, b, [y_1, y_1+\delta], \dots, [y_n, y_n+\delta])}{\mathbb{P}(Y_1 \in [y_1, y_1+\delta], \dots, Y_n \in [y_n, y_n+\delta])} \end{aligned} \tag{7}$$

where $f_{\mu,\sigma,Y_1,\dots,Y_n}$ is the joint distribution w.r.t. (μ, σ^2) and $(Y_1, \dots, Y_n)$ defined as:

$$\begin{aligned} & f_{\mu,\sigma,Y_1,\dots,Y_n}(a, b, [y_1, y_1+\delta], \dots, [y_n, y_n+\delta]) \\ &:= \lim_{\epsilon\to 0} \frac{\mathbb{P}(\mu \in [a, a+\epsilon], \sigma \in [b, b+\epsilon], Y_1 \in [y_1, y_1+\delta], \dots, Y_n \in [y_n, y_n+\delta])}{\epsilon^2}. \end{aligned}$$

Again we use the same small-sized interval $\epsilon = \epsilon_1 = \epsilon_2$ for illustrative purpose. Applying the definition of condition probability into (7), we get:

$$\begin{aligned} & f_{\mu,\sigma}(a, b|Y_1 \in [y_1, y_1+\delta], \dots, Y_n \in [y_n, y_n+\delta]) \\ &= \frac{f_{\mu,\sigma,Y_1,\dots,Y_n}(a, b, [y_1, y_1+\delta], \dots, [y_n, y_n+\delta])}{\mathbb{P}(Y_1 \in [y_1, y_1+\delta], \dots, Y_n \in [y_n, y_n+\delta])} \\ &= \frac{f_{\mu,\sigma}(a, b)\mathbb{P}(Y_1 \in [y_1, y_1+\delta], \dots, Y_n \in [y_n, y_n+\delta]|\mu = a, \sigma = b)}{\mathbb{P}(Y_1 \in [y_1, y_1+\delta], \dots, Y_n \in [y_n, y_n+\delta])} \\ &\approx \frac{f_{\mu,\sigma}(a, b)f(y_1, \dots, y_n|\mu = a, \sigma = b)\delta^n}{f(y_1, \dots, y_n)\delta^n} \end{aligned}$$

where the approximation is because we assume a very small δ. Notice that $f(y_1, \dots, y_n)$ is irrelevant to $(\mu, \sigma) = (a, b)$. So, the optimal estimator can be written as:

$$(\hat{\mu}_{\mathsf{opt}}, \hat{\sigma}^2_{\mathsf{opt}}) = \arg\max_{a,b\in\mathbb{R}} f_{\mu,\sigma}(a, b)f(y_1, \dots, y_n|\mu = a, \sigma = b). \tag{8}$$

As we assume no prior knowledge on $f_{\mu,\sigma}(a, b)$, as in the Bernoulli case, one natural assumption that we can make is:

$$f_{\mu,\sigma}(a, b) \text{ is uniformly distributed.} \tag{9}$$

Maximum Likelihood Estimator (MLE)

Since $f_{\mu,\sigma}(a, b)$ does not change w.r.t. (a, b) under the uniform distribution assumption (9), the optimal estimator can be simplified as:

$$(\hat{\mu}_{\mathsf{opt}}, \hat{\sigma}_{\mathsf{opt}}) = (\hat{\mu}_{\mathsf{ML}}, \hat{\sigma}_{\mathsf{ML}}) = \arg\max_{a,b\in\mathbb{R}} \underbrace{f(y_1, \dots, y_n|\mu = a, \sigma = b)}_{\text{likelihood}}. \tag{10}$$

Again, as in the Bernoulli case, we see that the optimal estimator is MLE.

MLE derivation

Let us now derive the MLE (10). Using the fact that $\{Y_i\}_{i=1}^n$ is i.i.d. $\sim \mathcal{N}(a, b^2)$ given $(\mu, \sigma) = (a, b)$, we get:

$$\begin{aligned} f(y_1, \ldots, y_n | \mu = a, \sigma = b) &= \prod_{i=1}^n f(y_i | \mu = a, \sigma = b) \\ &= \frac{1}{(\sqrt{2\pi} b)^n} e^{-\frac{1}{2b^2}\sum_{i=1}^n (y_i - a)^2}. \end{aligned}$$

The above expression is a bit ugly, containing a complicated term in the exponent. There is a conventional technique which allows us to make the expression cleaner. That is, to employ an increasing function, $\log(\cdot)$. Taking the log function, we can simplify the complicated term while preserving the maximizer. Taking the log function in the RHS of the above, we obtain:

$$\log\left(\frac{1}{(\sqrt{2\pi} b)^n} e^{-\frac{1}{2b^2}\sum_{i=1}^n (y_i - a)^2} \right) = -n \log(\sqrt{2\pi} b) - \frac{1}{2b^2} \sum_{i=1}^n (y_i - a)^2. \tag{11}$$

Taking this instead of the objective function in (10), we get:

$$(\hat{\mu}_{\mathsf{ML}}, \hat{\sigma}_{\mathsf{ML}}) = \arg \max_{a, b \in \mathbb{R}} \underbrace{-n \log(\sqrt{2\pi} b) - \frac{1}{2b^2} \sum_{i=1}^n (y_i - a)^2}_{=: \mathcal{L}(a,b)} \tag{12}$$

where $\mathcal{L}(a, b)$ is called the *log likelihood function.*

Now how to find the maximizer in (12)? From the course on Calculus or elsewhere (or from your intuition), you may hear and/or learn (or feel) that the stationary point in which the derivative is zero is often the maximizer. It turns out the above is indeed the case: the maximizer occurs at the stationary point. Actually, if you want to be convinced with rigour, you should have some background on *convex optimization* which you may hear of very often, yet you may not be familiar with. Here we will simply trust in the above statement, searching for the stationary point. If you want to know more in depth, you may want to take an introductory course on convex optimization: EE424 Introduction to Optimization. Don't worry about the final exam on this matter. We will not ask for any rigorous explanation as to why the maximizer occurs at the stationary point.

Relying upon the statement, we search for the stationary point by taking the derivative w.r.t. a and b:

$$\begin{aligned} \frac{d}{da} \mathcal{L}(a, b) &= -\frac{1}{b^2} \sum_{i=1}^n (y_i - a) \cdot (-1) \\ \frac{d}{db} \mathcal{L}(a, b) &= -\frac{n}{\sqrt{2\pi} b} \cdot \sqrt{2\pi} + \frac{1}{b^3} \sum_{i=1}^n (y_i - a)^2. \end{aligned} \tag{13}$$

Equating them to 0, we obtain the stationary point (a^*, b^*) as below:

$$\begin{aligned} \frac{d}{da} \mathcal{L}(a, b) \Big|_{a = a^*, b = b^*} &= \frac{1}{b^{*2}} \sum_{i=1}^n (y_i - a^*) = 0 & \longrightarrow a^* &= \frac{\sum_{i=1}^n y_i}{n}; \\ \frac{d}{db} \mathcal{L}(a, b) \Big|_{a = a^*, b = b^*} &= -n + \frac{1}{b^{*2}} \sum_{i=1}^n (y_i - a^*)^2 = 0 & \longrightarrow b^{*2} &= \frac{1}{n} \sum_{i=1}^n (y_i - a^*)^2. \end{aligned}$$

This solution is for a certain realization $\{y_i\}_{i=1}^n$. So when the random process $\{Y_i\}_{i=1}^n$ is fed into the estimator, the ML estimator reads:

$$\hat{\mu}_{\mathsf{ML}} = \frac{\sum_{i=1}^n Y_i}{n}, \qquad \hat{\sigma}^2_{\mathsf{ML}} = \frac{1}{n}\sum_{i=1}^n (Y_i - \hat{\mu}_{\mathsf{ML}})^2. \tag{14}$$

Notice that the first is exactly the *sample mean*, and the second is the average of $(Y_i - \hat{\mu}_{\mathsf{ML}})^2$, so it can be interpreted as the *sample variance*. So it makes quite an intuitive sense.

MLE in the limit of n

As in the Bernoulli case, one natural question arises. Do $\hat{\mu}_{\mathsf{ML}}$ and $\hat{\sigma}^2_{\mathsf{ML}}$ converge to the ground truths μ and σ^2, respectively, in the limit of n? To figure this out, let's invoke the LLN again. Applying the LLN to $\hat{\mu}_{\mathsf{ML}}$,

$$\hat{\mu}_{\mathsf{ML}} = \frac{\sum_{i=1}^n Y_i}{n} \overset{\text{in prob}}{\longrightarrow} \mathbb{E}[Y_i] = \mu \qquad \text{as } n \rightarrow \infty. \tag{15}$$

So the same mean converges to the true mean.

To see the convergence of $\hat{\sigma}^2_{\mathsf{ML}}$, we first re-write it as:

$$\begin{aligned} \hat{\sigma}^2_{\mathsf{ML}} &= \frac{1}{n}\sum_{i=1}^n (Y_i - \hat{\mu}_{\mathsf{ML}})^2 \\ &= \frac{1}{n}\sum_{i=1}^n Y_i^2 - \hat{\mu}^2_{\mathsf{ML}} \end{aligned} \tag{16}$$

where the 2nd equality is due to $\hat{\mu}_{\mathsf{ML}} = \frac{\sum_{i=1}^n Y_i}{n}$. Consider the 1st term in the second line of the above. Since $\{Y_i\}_{i=1}^n$ is i.i.d., its square counterpart $\{Y_i^2\}_{i=1}^n$ is also i.i.d. Why? For independent random variables, say X and Y, any functions, say $f(X)$ and $g(Y)$, are independent. If you are not convinced, please check. Now we wish to apply the LLN to $\{Y_i^2\}_{i=1}^n$. To this end, we first compute:

$$\mathbb{E}[Y_i^2] = \sigma^2 + (\mathbb{E}[Y_i])^2 = \sigma^2 + \mu^2. \tag{17}$$

We also need to check the finiteness of $\mathsf{Var}(Y_i^2)$. To figure this out, consider:

$$\mathsf{Var}(Y_i^2) = \mathbb{E}[Y_i^4] - (\sigma^2 + \mu^2)^2 \tag{18}$$

In fact, the computation of $\mathbb{E}[Y_i^4]$ is complicated, but it is doable. It turns out $\mathbb{E}[Y_i^4] = 3\sigma^4 + 6\mu^2\sigma^2 + \mu^4$. Putting this into the above,

$$\mathsf{Var}(Y_i^2) = \mathbb{E}[Y_i^4] - (\sigma^2 + \mu^2)^2 = 2\sigma^4 + 4\mu^2\sigma^2. \tag{19}$$

Hence, it is indeed finite. As you may guess, the exact computation of the variance is not that important. The only thing that matters is to check the finiteness. So don't be bothered by the exact computation of $\mathbb{E}[Y_i^4]$. Just remember that it is finite.

Applying the LLN to $\{Y_i^2\}_{i=1}^n$ together with (17),

$$\frac{\sum_{i=1}^n Y_i^2}{n} \overset{\text{in prob}}{\longrightarrow} \mathbb{E}[Y_i^2] = \sigma^2 + \mu^2. \tag{20}$$

Now remember what you learned from the course on Calculus. Suppose $\{a_i\}_{i=1}^{\infty}$ and $\{b_i\}_{i=1}^{\infty}$ are *convergent* sequences. Then, the limit of the sum of the two sequences converges to the sum of the two individual limits:

$$\lim_{n\to\infty}(a_n + b_n) = \lim_{n\to\infty} a_n + \lim_{n\to\infty} b_n. \tag{21}$$

Also, for any non-strange functions including the square function, the limit of the function of the *convergent* sequence is the same as the function of the limit. For instance,

$$\lim_{n\to\infty} a_n^2 = \left(\lim_{n\to\infty} a_n\right)^2. \tag{22}$$

These look intuitive. If you don't remember the proof of the above, you may want to try it using the definition of *convergence*. If you forget about the definition of convergence, review the relevant content in Calculus or search for wikipedia. On the other hand, if you feel comfortable in accepting the above without the proof, you may not try it at all. That is also okay, unless you wish to something hardcore-math in your future career. In reality, many facts like (21) and (22) coincide with our intuition, so forgetting about such hardcore-looking math is okay. Don't worry about the final exam on this hardcore math. You will not be asked to prove such convergence – this is not a course on *real analysis*, although it is relevant.

A side note: You know what? I was very annoyed by hardcore-looking math during my undergraduate. No intuition. No motivation. Too many definitions which are ugly-looking and highly non-intuitive. Even worse, no one explained to me about the rationale behind them. This is exactly why I decided not be a mathematician.

It turns out the convergence facts ((21) and (22)) also hold for the *convergence in probability*. Again, don't worry about the proof. Applying these to (16) together with (15), we get:

$$\sigma^2_{\mathsf{ML}} = \frac{1}{n}\sum_{i=1}^{n} Y_i^2 - \hat{\mu}^2_{\mathsf{ML}} \overset{\text{in prob}}{\longrightarrow} (\sigma^2 + \mu^2) - \mu^2 = \sigma^2. \tag{23}$$

As expected, σ^2_{ML} converges to the ground truth σ^2.

Look ahead

In Part II, we have thus far studied several concepts and key principles: (i) random processes, Bernoulli process, Gaussian process, Markov process; (ii) MAP principle in the context of the inference problem for cancer prediction; (iii) ML principle and its application to Bernoulli and Gaussian parameter estimation; (iv) Law of Large Numbers.

As mentioned earlier, prior to embarking on Part III (dedicated to applications), we need to study one more important theorem that plays a crucial role for Gaussian noise modeling. That is, the *Central Limit Theorem (CLT)*. So next time, we will investigate the CLT.

EE210 Probability and Introductory Random Processes
KAIST, Spring 2021

May 4, 2021
Changho Suh (chsuh@kaist.ac.kr)

Lecture 17: Central Limit Theorem (CLT)

Recap

Last time, we have studied the MLE for one prominent problem: Gaussian parameter estimation. As per the mean and variance (μ, σ^2), the i.i.d. Gaussian samples $\{Y_i\}_{i=1}^n$ are generated and fed into the estimation block, as illustrated in Fig. 1. Assuming no prior knowledge on the statistics of the parameters, we showed that the optimal estimate is equivalent to the ML estimate:

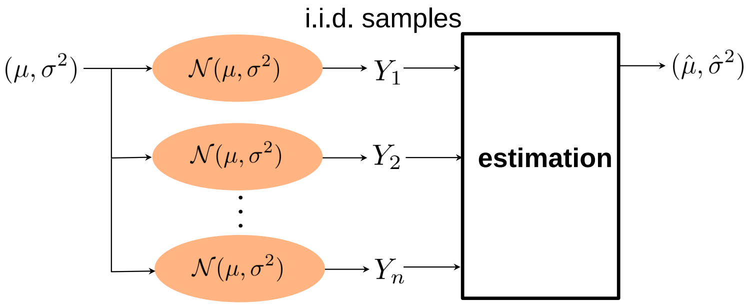

i.i.d. samples
(μ,σ^2)
$\mathcal{N}(\mu,\sigma^2)$
$\mathcal{N}(\mu,\sigma^2)$
$\mathcal{N}(\mu,\sigma^2)$
Y_1
Y_2
Y_n
estimation
$(\hat{\mu},\hat{\sigma}^2)$

Figure 1: Gaussian parameter estimation.

$$\begin{aligned}(\hat{\mu}_{\mathsf{opt}}, \hat{\sigma}^2_{\mathsf{opt}}) = (\hat{\mu}_{\mathsf{ML}}, \hat{\sigma}^2_{\mathsf{ML}}) &= \arg\max_{a,b\in\mathbb{R}} \underbrace{f(y_1,\ldots,y_n|\mu=a,\sigma=b)}_{\text{likelihood}} \\ &= \arg\max_{a,b\in\mathbb{R}} \frac{1}{(\sqrt{2\pi}b)^n} e^{-\frac{1}{2b^2}\sum_{i=1}^n (y_i-a)^2} \\ &= \arg\max_{a,b\in\mathbb{R}} -n\log(\sqrt{2\pi}b) - \frac{1}{2b^2}\sum_{i=1}^n (y_i-a)^2\end{aligned} \tag{1}$$

where the 2nd equality follows from the fact that $\{Y_i\}_{i=1}^n$ is i.i.d. $\sim \mathcal{N}(a, b^2)$ given $(\mu,\sigma)=(a,b)$; and the last equality is because taking the increasing function $\log(\cdot)$ does not alter the optimal solution. Finding the *stationary* point by taking the derivative, we showed that the maximizer matches with our intuition, *sample mean* & *sample variance*:

$$\hat{\mu}_{\mathsf{ML}} = \frac{\sum_{i=1}^n Y_i}{n}, \quad \hat{\sigma}^2_{\mathsf{ML}} = \frac{1}{n}\sum_{i=1}^n (Y_i - \hat{\mu}_{\mathsf{ML}})^2.$$

Using the Law of Large Numbers (LLN), we also proved that the ML estimate converges to the ground truth (in probability) as the number n of observations tends to infinity: as $n \to \infty$,

$$\begin{aligned}\hat{\mu}_{\mathsf{ML}} &= \frac{\sum_{i=1}^n Y_i}{n} \xrightarrow{\text{in prob}} \mathbb{E}[Y_i] = \mu, \\ \hat{\sigma}^2_{\mathsf{ML}} &= \frac{1}{n}\sum_{i=1}^n (Y_i - \hat{\mu}_{\mathsf{ML}})^2 = \frac{1}{n}\sum_{i=1}^n Y_i^2 - \hat{\mu}^2_{\mathsf{ML}} \xrightarrow{\text{in prob}} (\sigma^2+\mu^2) - \mu^2 = \sigma^2.\end{aligned}$$

At the end of the last lecture, I mentioned that prior to Part III (dedicated to applications), we need to study one more theorem that plays a crucial role for Gaussian modeling of *noise* that often appears in many contexts: the *Central Limit Theorem (CLT)*. This is the topic of this lecture.

Today's lecture

Today we will investigate the CLT in depth. This lecture consists of three parts. We will first make the precise statement of the CLT. Next we will study two claims that play an important role in the proof: (i) Claim #1, concerning the sum of multiple *independent continuous* random variables; (ii) Claim #2, enabling a simpler expression of a complicated-looking pdf via *Laplace transform*. Finally we will use the two claims to prove the theorem.

Rough statement of CLT

Let $\{X_i\}_{i=1}^{\infty}$ be an i.i.d. process with mean $\mathbb{E}[X_i] = \mu$ and finite variance $\mathsf{Var}(X_i) = \sigma^2 < \infty$. A rough statement of the CLT is that in the limit of $n \to \infty$,

$X_1 + X_2 + \cdots + X_n$ converges to a continuous r.v. with the *Gaussian* distribution.

Here the mean of the sum should read $\mathbb{E}[X_1 + \cdots + X_n] = n\mathbb{E}[X_1] = n\mu$ and its variance is:

$$\begin{aligned} \mathsf{Var}(X_1 + \cdots + X_n) &= \mathsf{Var}(X_1) + \cdots + \mathsf{Var}(X_n) \\ &= n\mathsf{Var}(X_1) = n\sigma^2 \end{aligned} \tag{2}$$

where the 1st equality is due to the independence of $\{X_i\}_{i=1}^{n}$. Perhaps one surprising fact in the CLT is that the density of the converged random variable is *Gaussian* no matter what the initial distribution of the original random process is.

Precise statement of CLT

The above statement is a bit rough in the following two reasons. First, the converged random variable has *infinite* mean and variance. Second, the meaning of the convergence to another random variable is not rigorously defined. To make the converged random variable have *finite* mean and variance, people often do *normalization* (subtracting the mean and dividing it by the standard deviation):

$$Z_n := \frac{X_1 + X_2 + \cdots + X_n - n\mu}{\sqrt{n\sigma^2}}. \tag{3}$$

This way, we get: $\mathbb{E}[Z_n] = 0$ and $\mathsf{Var}(Z_n) = 1$. The precise statement of the CLT is then:

$$Z_n = \frac{X_1 + X_2 + \cdots + X_n - n\mu}{\sqrt{n\sigma^2}} \xrightarrow{\text{in distribution}} Z \sim \mathcal{N}(0,1). \tag{4}$$

Here what it means by "convergence in distribution" is that the cumulative density distribution of Z_n is the same as that of Z in the limit:

$$\lim_{n\to\infty} F_{Z_n}(z) = \int_{-\infty}^{z} \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}t^2} dt \qquad \forall z \in \mathbb{R}. \tag{5}$$

A side note: There is a reason why the convergence in distribution is defined w.r.t. the cdf instead of the pdf. The reason is that there are some tricky yet unusual situations in which the cdf converges while the pdf does not. But for illustrative simplicity, we are going to use the pdf instead of the cdf from now on. If you want to know more in detail, unfortunately you should take a hardcore-math course like "measure theory" which has been driving me to sit in the Department of Electrical Engineering.

Remember the convergence in *probability*. It concerns the *probabilistic* comparison between a *random* variable and a *deterministic* converged quantity. On the other hand, the convergence in *distribution* is about comparison with regard to distribution between two random variables (Z_n and Z in the above case). So you may now understand why we have multiple definitions of convergence when dealing with random processes (not deterministic sequences).

Equivalent statement of CLT

There is a simpler and equivalent expression of (4) which I prefer and we will prove in this lecture. The expression is based on the following observation:

$$\frac{X_1 + X_2 + \cdots + X_n - n\mu}{\sqrt{n\sigma^2}} = \underbrace{\frac{X_1 - \mu}{\sqrt{n\sigma^2}}}_{=:X_1'} + \underbrace{\frac{X_2 - \mu}{\sqrt{n\sigma^2}}}_{=:X_2'} + \cdots + \underbrace{\frac{X_n - \mu}{\sqrt{n\sigma^2}}}_{=:X_n'}. \tag{6}$$

Here X_i' is just a shifted version of X_i. So $\{X_i'\}_{i=1}^{\infty}$ is also i.i.d. yet with mean $\mathbb{E}[X_i'] = 0$ and variance

$$\begin{aligned}\mathsf{Var}(X_i') &= \mathsf{Var}\left(\frac{X_1 - \mu}{\sqrt{n\sigma^2}}\right) \\ &= \frac{1}{n\sigma^2}\mathsf{Var}\left(X_1 - \mu\right) \\ &= \frac{\sigma^2}{n\sigma^2} = \frac{1}{n}\end{aligned}$$

where the 2nd equality is due to $\mathsf{Var}(cX) = \mathbb{E}[(cX)^2] - (\mathbb{E}[cX])^2 = c^2\mathsf{Var}(X)$ for any constant c and a random variable X; and the 3rd equality is because $\mathsf{Var}(X_1 - \mu) = \mathsf{Var}(X_1) = \sigma^2$ (Why a constant shift does not alter the variance? Think about it). Hence, in terms of $\{X_i'\}_{i=1}^{\infty}$, the CLT says:

$$X_1' + X_2' + \cdots + X_n' \overset{\text{in dist}}{\longrightarrow} Z \sim \mathcal{N}(0,1). \tag{7}$$

For notational simplicity, let's re-use the simpler non-prime notation: $\{X_i\}_{i=1}^{\infty}$. With the simpler notation, the CLT then says:

$$Z_n := X_1 + X_2 + \cdots + X_n \overset{\text{in dist}}{\longrightarrow} Z \sim \mathcal{N}(0,1). \tag{8}$$

where $\{X_i\}_{i=1}^{\infty}$ is i.i.d. with $\mathbb{E}[X_i] = 0$ and variance $\mathsf{Var}(X_i) = \frac{1}{n}$. In this lecture, we will prove the equivalent statement (8), i.e.,

$$\lim_{n\to\infty} f_{Z_n}(z) = \frac{1}{\sqrt{2\pi}}e^{-\frac{1}{2}z^2} \qquad \forall z \in \mathbb{R}. \tag{9}$$

It turns out there are two important claims that play a crucial role in streamlining the proof. So we will first investigate the two claims and then use them to complete the proof (9).

Claim #1

Notice in (8) that there are many *sums* in relating Z_n to $\{X_i\}_{i=1}^{n}$. At the end of the day, we are interested in figuring out the pdf of Z_n summed by many X_i's. The first claim is about the pdf of the sum of multiple independent random variables. For illustrative purpose, consider only two random variables, say X and Y. Let $Z = X + Y$. Remember the *discrete* random variable case where the pmf of the sum of two independent discrete random variables is expressed as the *convolution* of the individual pmfs. Claim #1 says that it holds also for *continuous* random variables:

$$\text{(Claim \#1): } f_Z(z) = (f_X * f_Y)(z) := \int_{-\infty}^{\infty} f_X(x) f_Y(z-x) dx. \tag{10}$$

The proof of this is not that difficult, so we will omit the proof here. Instead you will have a chance to prove it in PS5.

Claim #2

Recall the interested random variable Z_n:

$$Z_n = X_1 + X_2 + \cdots + X_n.$$

Applying Claim#1 (10) many times, we obtain:

$$f_{Z_n}(z) = (f_{X_1} * f_{X_2} * \cdots * f_{X_n})(z). \tag{11}$$

We have many convolutions. The convolution formula is complicated, as the meaning of the word "convoluted" suggests. So the expression of $f_{Z_n}(z)$ is very much complicated. This is where Claim #2 kicks in. What Claim #2 says is that the convolutions can be significantly simplified in the *Laplace transform* domain. To understand what it means in detail, let us consider the above setting in which there are only two independent random variables, X and Y, and $Z = X + Y$. Claim #2 says that the Laplace transform $F_Z(s)$ of $f_Z(z)$ can be expressed as the product of the individual Laplace transforms $(F_X(s), F_Y(s))$ w.r.t. $(f_X(x), f_Y(y))$:

$$\text{(Claim \#2): } F_Z(s) = F_X(s) F_Y(s) \tag{12}$$

where the Laplace transform is defined as:

$$F_Z(s) := \int_{-\infty}^{+\infty} e^{-sz} f_Z(z) dz. \tag{13}$$

The proof of this is also easy, relying only upon the "change-of-variable" technique that we exercised with several times earlier. Hence, we skip the proof as well. But you can check it in PS5.

Setup for the proof of CLT

We are now ready to prove the CLT (8). Applying Claim #2 into (11) many times (more precisely, $n-1$ times), we get:

$$\begin{aligned} F_{Z_n}(s) &= F_{X_1}(s) F_{X_2}(s) \cdots F_{X_n}(s) \\ &= [F_{X_1}(s)]^n \end{aligned} \tag{14}$$

where the 2nd equality is because each of $\{X_i\}_{i=1}^{\infty}$ is identically distributed. For simplicity, let's not worry about a very rare and practically-irrelevant case where the Laplace transform does not exist.

Now how to proceed with (14)? More precisely, how to compute $F_{Z_n}(s)$ with what we know: $\mathbb{E}[X_1] = 0$ and $\mathbb{E}[X_1^2] = \frac{1}{n}$? It turns out these moment information (1st and 2nd moments) play a role to compute. Here a key observation is that these moments appear as coefficients in the *Taylor series expansion* of $F_{X_1}(s)$. And it turns out this expansion leads to an explicit expression for $F_{Z_n}(s)$, thereby hinting the pdf of Z_n. To see this, let us try to obtain the Taylor series expansion of $F_{X_1}(s)$.

Taylor series expansion

To this end, we first need to compute the kth derivative of $F_{X_1}(s)$:

$$\begin{aligned} F_{X_1}^{(k)}(s) &:= \frac{d^k F_{X_1}(s)}{ds^k} \\ &= \frac{d^k}{ds^k}\left\{\int_{-\infty}^{\infty} e^{-sx} f_{X_1}(x)dx\right\} \\ &= \int_{-\infty}^{\infty} (-1)^k x^k e^{-sx} f_{X_1}(x)dx \end{aligned} \tag{15}$$

where the 2nd equality is due to the definition of the Laplace transform: $F_{X_1}(s) := \int_{-\infty}^{+\infty} e^{-sx} f_{X_1}(x)dx$; and the 3rd equality is because of the interchangeability of integration and differentiation (let's not worry about a very weird situation where it is not interchangeable). Applying the Taylor expansion at $s = 0$, we get:

$$F_{X_1}(s) = \sum_{k=0}^{\infty} \frac{F_{X_1}^{(k)}(0)}{k!} s^k. \tag{16}$$

Plugging $s = 0$ into (15), we get:

$$F_{X_1}^{(k)}(0) = \int_{-\infty}^{\infty} (-1)^k x^k f_{X_1}(x)dx = (-1)^k \mathbb{E}[X_1^k].$$

This together with (16) yields:

$$F_{X_1}(s) = \sum_{k=0}^{\infty} \frac{(-1)^k \mathbb{E}[X_1^k]}{k!} s^k. \tag{17}$$

Applying $\mathbb{E}[X_1] = 0$ and $\mathbb{E}[X_1^2] = \frac{1}{n}$ to the above, we obtain:

$$F_{X_1}(s) = 1 + \frac{1}{2n}s^2 + \sum_{k=3}^{\infty} \frac{(-1)^k \mathbb{E}[X_1^k]}{k!} s^k. \tag{18}$$

Now what can we say about $\mathbb{E}[X_1^k]$ for $k \geq 3$ in the above? Notice that $\mathbb{E}[X_1^k] = \mathbb{E}[(X_1^2)^{\frac{k}{2}}]$ and $\mathbb{E}[X_1^2] = \frac{1}{n}$ scales like $\frac{1}{n}$. It turns out this leads to the fact that $\mathbb{E}[(X_1^2)^{\frac{k}{2}}]$ decays like $\frac{1}{n^{k/2}}$, which exhibits a faster decaying rate for $k \geq 3$, as compared to $\frac{1}{n}$. It turns out this scaling yields:

$$\begin{aligned} \lim_{n\to\infty} F_{Z_n}(s) &= \lim_{n\to\infty} \left(1 + \frac{1}{2n}s^2 + \sum_{k=3}^{\infty} \frac{(-1)^k \mathbb{E}[X_1^k]}{k!} s^k\right)^n \\ &= \lim_{n\to\infty} \left(1 + \frac{1}{2n}s^2\right)^n \\ &= e^{\frac{s^2}{2}} \end{aligned} \tag{19}$$

where the last equality is due to the fact that

$$e^x = \lim_{n\to\infty} \left(1 + \frac{x}{n}\right)^n. \tag{20}$$

Here the rigorous proof for the 2nd equality in (19) is omitted. Instead we only provided the intuition behind that: $\mathbb{E}[(X_1^2)^{\frac{k}{2}}]$ decays like $\frac{1}{n^{k/2}}$, exhibiting a much faster decaying rate than $\frac{1}{n}$, for $k \geq 3$. In fact, the rigorous proof requires hardcore-math which may incur too much distraction without giving any insights. That's why I skipped it. Also, don't worry about the final exam. You will not be asked to provide a rigorous proof on this.

Proof of CLT

From (19), what can we say about $f_{Z_n}(z)$ in the limit of n? To figure this out, we need to do the Laplace *inverse* transform:

$$f_{Z_n}(z) = \text{InverseLaplace}(F_{Z_n}(s))(z) := \frac{1}{2\pi i} \lim_{T\to\infty} \int_{\mathsf{Re}(s)-iT}^{\mathsf{Re}(s)+iT} F_{Z_n}(s) e^{sz} ds \tag{21}$$

where $\mathsf{Re}(s)$ denotes the real-component value in s. But you may feel headache because it looks very much complicated. So instead we will do the reverse engineering: guess-&-check. A good thing about the Laplace transform is that it is an one-to-one mapping. Also in PS4, you have already computed:

$$\text{LaplaceTransform}\left(\frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}}\right) = e^{\frac{s^2}{2}}.$$

This together with (19) gives:

$$\lim_{n\to\infty} f_{Z_n}(z) = \frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}}, \quad z \in \mathbb{R}. \tag{22}$$

This is indeed the Gaussian distribution with mean 0 and variance 1. So this completes the proof of the CLT.

Look ahead

In Part I, we have studied numerous concepts in probability: sample space, events, conditional probability, total probability law, independence, random variables. In Part II, we have studied some deeper concepts and key principles: random processes, MAP principle, ML principle, Law of Large Numbers and Central Limit Theorem.

The goal of Part III is to demonstrate the role of such concepts and principles in the context of the following three killer applications: (i) communication; (ii) machine learning; and (iii) speech recognition. Next time, we will first focus on the communication application.

EE210 Probability and Introductory Random Processes May 6, 2021
KAIST, Spring 2021 Changho Suh (chsuh@kaist.ac.kr)

Lecture 18: Communication: A probabilistic modeling of noise

Recap

Last time, we have explored an important theorem which forms the basis of the Gaussian modeling of many interested random quantities: the *Central Limit Theorem (CLT)*. It is about an i.i.d. random process $\{X_i\}_{i=1}^{\infty}$. For illustrative purpose, we considered a simple yet generalizable setup: $\mathbb{E}[X_i] = 0$ and $\mathsf{Var}(X_i) = \frac{1}{n}$. In this setup, the CLT says:

$$Z_n := X_1 + X_2 + \cdots + X_n \overset{\text{in dist}}{\longrightarrow} Z \sim \mathcal{N}(0,1). \tag{1}$$

where the meaning of the description $\overset{\text{in dist}}{\longrightarrow}$ is:

$$\lim_{n\to\infty} f_{Z_n}(z) = \frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}z^2} \qquad \forall z \in \mathbb{R}. \tag{2}$$

We used two claims to prove the CLT. Using the 1st claim, we expressed the pdf of Z_n as the convolution of many pdfs w.r.t. X_i's: $f_{Z_n}(z) = (f_{X_1} * f_{X_2} * \cdots * f_{X_n})(z)$. Employing the 2nd claim that enables a succinct representation via Laplace transform, we obtained: $F_{Z_n}(s) = [F_{X_1}(s)]^n$. We then used Taylor series expansion to show that:

$$\lim_{n\to\infty} F_{Z_n}(s) = \lim_{n\to\infty} \left(1 + \frac{1}{2n}s^2\right)^n = e^{\frac{s^2}{2}}. \tag{3}$$

Lastly, exploiting the one-to-one mapping property of Laplace transform together with

$$\text{LaplaceTransform}\left(\frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}}\right) = e^{\frac{s^2}{2}},$$

we finally proved the CLT (2).

Now let's recall what we have learned so far. In Part I, we have studied many concepts: sample space, events, conditional probability, independence, random variables, and Gaussian distribution. In Part II, we have investigated a couple of deeper concepts and key principles: (i) random processes and three prominent examples; (ii) MAP and ML principles; (iii) Law of Large Numbers; (iv) Central Limit Theorem. The goal of Part III is to demonstrate the role of such contents in the context of the three killer applications: (i) communication; (ii) machine learning; (iii) speech recognition.

Today's lecture

Today we will focus on the first communication application to figure out the connection with probability. Specifically what we are going to do are five folded. First of all, we will review the definition of communication and then introduce the architecture of *digital* communication that we will put a special emphasis on. Second, we will investigate an *uncertain* entity that arises in communication and therefore the one that is intimately related to probability. That is, *noise*. As mentioned earlier several times, the noise can be modeled as the famous *Gaussian* random variable. In the 3rd and 4th parts, we will study physical properties of the noise and then develop a mathematical model based on physics. Lastly we will build upon the mathematical framework to show that the probability distribution of the noise is Gaussian.

Digital communication

Recall the definition of communication that we studied in Lecture 1. Communication is the transfer of information from one end to the other end. Here the one end is called the *transmitter*, and the other end is called the *receiver*. Something that lies in between is a physical medium, called the *channel*.

Broadly speaking, there are two kinds of communication depending on the type of *information source* that we wish to transmit: (i) *analog* communication in which the information source is of *raw* type, like sound waveforms, images, and texts; (ii) *digital* communication in which the information source is simply binary string (a sequence of binary digits), simply called the *bits*. In this course, we will focus on the digital communication, as it has laid the foundation of almost all the current communication systems.

The fact that digital communication has been dominating the communication system era is based on the following key finding by one genius scientist in the mid 1900s: *Claude E. Shannon*. What he showed is that any type of information source can be represented as bits without losing the meaning of the information. One simple example is an English text that comprises English letters. Here one key observation is that there are only a *finite* number of candidates that each letter can take on. This number is the total number of English alphabets, which is 26. Here we ignore any special characters such as space. From this, we see that $\lceil \log_2 26 \rceil = 5$ number of bits suffices to represent each letter. One can make a similar argument for any other type of information source.

Fig. 1 illustrates the architecture of digital communication that Shannon introduced. Bits are the information source that we wish to transmit. The transmitter has one white box which takes the bits as an input to yield a signal that would be transmitted over the channel. Shannon called the box the *encoder*. Similarly the receiver has another white box which takes a received signal to try to reconstruct the bits as perfectly as possible. The box is called the *decoder*.

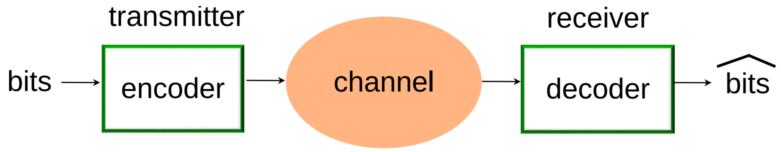


Figure 1: The architecture of digital communication.

Modem

Let me introduce one terminology that you may hear of in the context of communication. The physical world is definitely analog, so a transmitted signal that is fed into the channel should be a physical quantity, such as an electromagnetic signal, say an electrical voltage signal. So the encoder needs to *modulate* the digital information (bits) into an electrical voltage signal. Also a received signal (the channel output) is a voltage signal, so the decoder needs to *demodulate* the analog signal to reconstruct the bits. Hence, people often call the encoder/decoder simply the *modem*, highlighting “mo” and “dem” from modulator and demodulator, respectively. See Fig. 2.

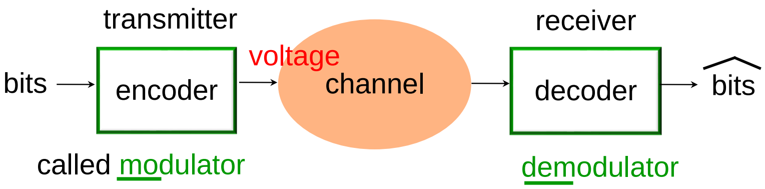


Figure 2: MODEM: MOdulator & DEMoluator.

A probabilistic model of the channel

One key feature of the channel is its uncertainty. The received voltage signal is not a deterministic function of the transmitted signal. Also the channel in most practical scenarios is of the form of addition: the received signal is the sum of the transmitted signal and an additive noise, as illustrated in Fig. 3. Hence, the additive noise, say Z, can be described by a random quantity.

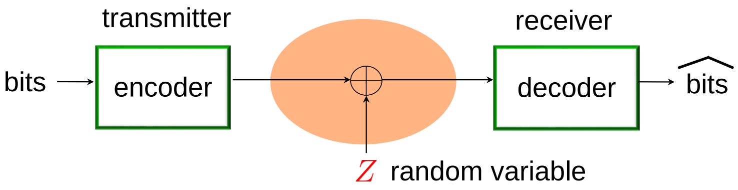


Figure 3: An additive noise channel.

In the language of probability, the random quantity is referred to as a *random variable*. More precisely, it is a *continuous* random variable, which is described by a probability density function. As mentioned in the beginning, people found that Z can be modeled as the famous *Gaussian* random variable. For the rest of this lecture, we will prove it based on physical properties of the noise.

Physics

Let us first invoke physics, by investigating how such noise occurs in interested communication systems. Here what I mean by interested communication systems is the one with electronic circuits, which I simply call electronic communication systems. In such *electronic* communication systems, a signal level is determined by its voltage. And according to the basic introductory course on circuits which many of you may be taking now: the voltage level is intimately related to the movement of electrons. The less electrons, the higher a voltage level.

In the early 1900s, it was discovered by a physicist, named John B. Johnson, that the behavior of electrons contributes to inducing a noise that we cannot control over. The discovery was based on an interesting observation that electrons are *randomly agitated* by heat. This random agitation then makes the voltage level be out of control, as it may depend on a non-controllable factor which is the temperature. So the precise movement control of electrons is almost impossible in reality. Jonhson interpreted such undesirable random fluctuation as a major source of *noise*.

At that time, Johnson wanted to figure out the statistical behavior of the noise, but he could not do so because he was an experimental physicist, not being good at math. Instead he had a colleague in his workplace, Bell Labs, who is very smart and particularly strong at math. The colleague was Harry Nyquist. So he shared his experimental observation with Nyquist. As

Johnson expected, Nyquist could establish a mathematical theory for such noise to demystify its statistical behavior. The theory formed the basis of the Gaussian noise modeling that we will study in the sequel. At the time, the noise was named as the thermal noise, as it is dependent of the temperature.

In reality, the noise may also depend on device imperfections and/or measurement inaccuracies. But the major source of the randomness is due to the thermal noise. So we are going to focus on the mathematical theory of the thermal noise done by Nyquist.

Assumptions made on the thermal noise

The mathematical theory starts with making concrete assumptions on physical properties that the thermal noise respects. These are four folded.

1. As mentioned earlier, the noise is due to the random movement of electrons. And an electrical signal contains tons of electrons. So one natural assumption that one can make is: the additive noise is the overall consequence of *many additive "sub-noises"*. A natural follow-up assumption is that the number of sub-noises is infinity, which reflects the fact that there are tons of electrons in an electrical signal.

2. These electrons are known to be typically uncorrelated with each other. So the second assumption is: the sub-noises are *mutually independent*.

3. Also, it is known that there is no particularly dominating electron that affects the additive noise most significantly. So the third assumption that one can make as a simplified version of this finding is: each sub-noise contributes exactly the same amount of energy to the total energy in the additive noise.

4. Finally, the noise energy is typically not so big relative to the energy of an interested voltage signal. Obviously it does not blow up. So the last reasonable assumption is: the noise energy is *finite*.

A mathematical model based on the four assumptions

Let us express the above four assumptions in a mathematical language. To this end, we first introduce some mathematical notations to write the total additive noise Z as the infinite sum of n sub-noises, say $X_1, X_2, \ldots, X_n$:

$$Z = \lim_{n\to\infty} \underbrace{X_1 + X_2 + \cdots + X_n}_{=:Z_n}. \tag{4}$$

The mathematical expressions for the first and second assumptions are: $n \to \infty$ and $(X_1, \ldots, X_n)$ are *mutually independent*. A mathematical assumption that we will take w.r.t. the third assumption is: $(X_1, \ldots, X_n)$ are *identically distributed*. So it is the i.i.d. assumption.

Without loss of generality, one can assume that Z has zero-mean. Here what it means by "without loss of generality" is that the general case can be readily covered with some proper modification to the ground assumption that follows after the phrase. That's why here we say that there is no loss of generality. You may wonder why the zero-mean assumption can serve as the ground assumption. To see this, consider a general case wherein we have a bias on Z: $\mathbb{E}[Z] = \mu \neq 0$. We can then always subtract the bias from the received signal so that the mean of the *effective noise* is zero. More precisely, we subtract the bias μ from the received signal $Y = X + Z$ (where X denotes the transmitted signal) to obtain:

$$Y - \mu = X + (Z - \mu). \tag{5}$$

Here $Z-\mu$ can be interpreted as the *effective noise*, and this has indeed zero-mean. Some may wonder how we can figure out the mean μ. There is a very popular way to estimate μ based on the maximum likelihood (ML) principle that you learned. This will be explored in depth in PS6.

Due to the i.i.d. assumption, each sub-noise has the same amount of energy:

$$\mathbb{E}[X_i^2]=\mathbb{E}[X_j^2], \ \ \forall i,j=1,\ldots,n. \tag{6}$$

Here the energy is simply represented as the square of the voltage based on the fact that the energy is proportional to the voltage square. We ignore other factors like "resistance". The energy in the aggregated noise Z_n is

$$\begin{aligned}\mathbb{E}[Z_n^2] &= \mathsf{Var}(Z_n)\\ &= \mathsf{Var}(X_1)+\mathsf{Var}(X_2)+\cdots+\mathsf{Var}(X_n)\\ &= n\mathsf{Var}(X_1) =: \sigma^2\end{aligned} \tag{7}$$

where the 1st equality is due to $\mathbb{E}[Z_n]=0$ (the zero-mean assumption); and the 2nd and 3rd equalities come from the i.i.d. assumption.

Lastly consider the finite energy assumption. Denoting by σ^2 this finite energy, we see from (7) that the energy of the sub-noises must shrink as more and more sub-noises are added:

$$\mathbb{E}[X_i^2]=\frac{\sigma^2}{n}, \ \ i=1,\ldots,n. \tag{8}$$

The statistical behavior of the thermal noise in the limit of n

We are interested in the pdf of the random variable Z_n in the limit of n. Recall $Z_n=X_1+X_2+\cdots+X_n$ and the above assumptions are summarized as below:

$$\begin{aligned}&(X_1,\ldots,X_n) \text{ i.i.d.};\\ &\mathbb{E}[X_i]=0 \qquad \forall i;\\ &\mathsf{Var}(X_i)=\frac{\sigma^2}{n} \qquad \forall i.\end{aligned} \tag{9}$$

What does this remind you of? Yes, it is the *Central Limit Theorem.* The only distinction here is that we read $\mathsf{Var}(X_i)=\frac{\sigma^2}{n}$ instead of $\mathsf{Var}(X_i)=\frac{1}{n}$. This distinction leads to $\mathsf{Var}(Z)=\sigma^2$ instead of $\mathsf{Var}(Z)=1$. Hence, using the CLT, we get:

$$Z=\lim_{n\rightarrow\infty} Z_n \sim \mathcal{N}(0,\sigma^2). \tag{10}$$

Look ahead

In this lecture, we made some physics-inspired assumptions on the additive thermal noise to demonstrate that the additive noise can precisely be modelled as the Gaussian distribution. See Fig. 4. Next time, we will explore how to design the transmission and reception strategies under the Gaussian channel. In the process, we will show that the MAP principle plays a key role.

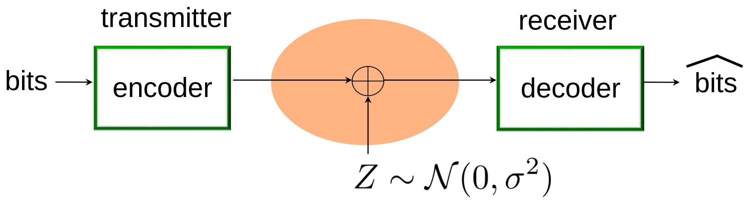


Figure 4: Additive Gaussian noise channel.

EE210 Probability and Introductory Random Processes
KAIST, Spring 2021

May 11, 2021
Changho Suh (chsuh@kaist.ac.kr)

Lecture 19: Communication: MAP principle

Recap

Last time we have studied a probabilistic modeling of the noise that appears as a source of uncertainty in communication. The noise is often added on top of a transmitted signal, as illustrated in Fig. 1. Due to the uncertain nature of the noise, it is described in terms of a *random variable*, say Z. Based on several physical properties of the noise, we showed that Z can be modeled as a *Gaussian* random variable. Specifically, the physical properties that we pointed out were: (i) the noise is a consequence of an aggregation of many "sub-noises"; (ii) there is almost no correlation across the sub-noises; (iii) there is no dominating sub-noise; and (iv) it has a finite energy. We then translated the properties into the following mathematical model:

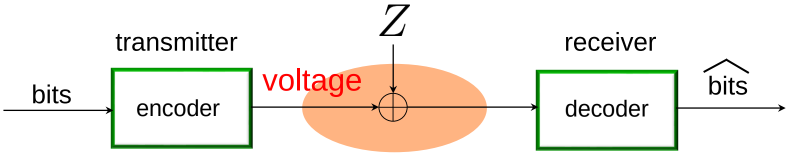


Figure 1: The additive Gaussian noise channel.

$$Z = \lim_{n\to\infty} \underbrace{X_1 + X_2 + \cdots + X_n}_{=:Z_n} \tag{1}$$

where $\{X_i\}_{i=1}^n$ are i.i.d. with $\mathbb{E}[X_i] = 0$ and $\mathsf{Var}(X_i) = \frac{\sigma^2}{n}$ for all i. Lastly applying the CLT into the above, we proved:

$$Z = \lim_{n\to\infty} Z_n \sim \mathcal{N}(0, \sigma^2). \tag{2}$$

Today's lecture

Today we will explore transmission and reception strategies under the additive Gaussian noise channel. One crucial point that I would like to point out in the process is that the *optimal receiver* is based on the MAP and ML principles that have been emphasized throughout. Specifically what we are going to do are four folded. We will first consider a simple yet wide-employed transmission scheme. Under the simple transmission, we will then guess a reasonably good reception scheme based solely upon intuition. Next we will derive the *optimal* receiver based on the MAP principle. Lastly we will show that under a reasonable assumption, the optimal decision rule reduces to the ML estimation and furthermore it coincides with the initial guess.

A simple transmission scheme for sending one bit

As a simple setting, we consider the case of sending only one bit, say B. Since the transmitted bit is *unknown* to the receiver, the interested bit B can be viewed as a binary *random variable.*

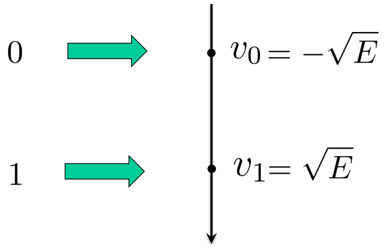


Figure 2: Pulse Amplitude Modulation (PAM): Mapping the value of one bit into one of the two possible voltage levels.

In an electronic communication system, a transmitted signal must be an analog voltage signal. So B should be modulated into a voltage signal. To this end, we consider a simple binary mapping; see Fig. 2. The case $B = 0$ is mapped to a voltage level v_0. Similarly v_1 corresponds to $B = 1$. How to choose v_0 and v_1? The choice is related to a communication budget given as a constraint in the system. Physically, the voltage transmitted corresponds to some energy being spent. According to the circuit theory that many of you guys are familiar with, the energy is proportional to the square of the voltage. For simplicity, let us assume that the energy spent in transmitting a voltage v Volts is exactly v^2 Joules. Under this assumption, a good choice of (v_0, v_1) might be the one that maximizes the distance $|v_0 - v_1|$ (the larger distance, the easier to distinguish the two levels), given the energy budget constraint $v_i^2 \leq E$ for $i = 0, 1$. This is an easy optimization problem wherein the solution is simply obtained at the boundary point: $v_0 = -\sqrt{E}$ and $v_1 = +\sqrt{E}$. Hence we take this choice, as illustrated in Fig. 2. Actually this is a prominent well-known scheme called the *Pulse Amplitude Modulation (PAM).* Notice that the bit information is reflected via perturbing (modulating) a voltage-pulse amplitude level.

Guess a reasonably good reception scheme

Under the above transmission strategy, what is a good reception scheme? To figure this out, let's first consider two main factors that affect a reception rule. The first is the *a priori probability*: $\mathbb{P}(B = 0)$ and $\mathbb{P}(B = 1)$. Why does this matter? You can easily see why if you think about one extreme case where we somehow knew (prior to the communication process) that the information bit is 0 for sure. In this case, we do not need to take a look at the received voltage at all. We can simply declare that the information bit is always 0, no matter what and whatsoever. In reality, however, we often times have no access to such prior information. In this case, what we can do is to simply assume that the information bit is *equally likely* to be 1 or 0. So here we will take this assumption:

$$\text{(Assumption)}: \mathbb{P}(B = 0) = \mathbb{P}(B = 1) = \frac{1}{2}. \tag{3}$$

The second factor is *noise statistics.* Knowing the statistical behavior of the noise will help the receiver make a decision. To see this, we consider the Gaussian noise of interest. The Gaussian noise is more likely to be near zero, as illustrated in Fig. 3. Here the height at a certain point of a bell-shaped curve indicates the frequency density of the occurrence at the point. In this case, your intuition says that a good receiver would be the one that picks the nearer of the two possible transmitted voltages as compared to the received voltage. In other words, if the received voltage is above the middle (0 in this case) of the two possible voltages ($\pm\sqrt{E}$), we declare $\hat{B} = 1$; otherwise, $\hat{B} = 0$. Actually this is an intuitive and famous rule, named the *Nearest Neighbor (NN)* rule. So one can image that the optimal decision rule in the Gaussian noise case might be the NN rule. It turns out this is indeed the case. For the rest of this lecture,

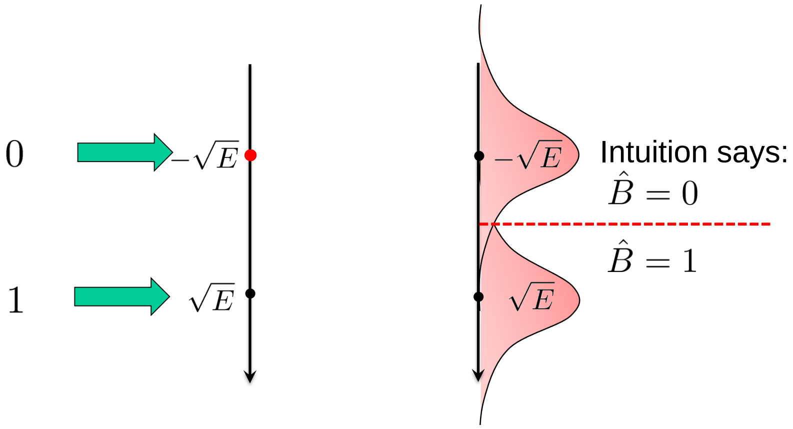


Figure 3: A guess about the optimal receiver.

we will prove this.

The optimal decision rule

Recall by definition that the optimal decision rule is the one that maximizes correct-decision probability. So we focus on the interested probability. Given a particular realization of the received signal, say $Y = y$, the correct-decision probability reads:

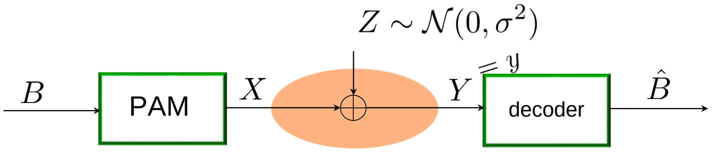


Figure 4: The optimal receiver is the one that maximizes $\mathbb{P}(B = \hat{B}|Y = y)$.

$$\mathbb{P}(B = \hat{B}|Y = y) \tag{4}$$

where $\hat{B}$ is the estimate of B; see Fig. 4. Hence, the optimal decision rule is *MAP*:

$$\hat{B}_{\mathsf{opt}} = \hat{B}_{\mathsf{MAP}} = \arg \max_{\hat{b}\in\{0,1\}} \mathbb{P}(B = \hat{b}|Y = y). \tag{5}$$

MAP derivation

Let us massage the MAP solution (5) to obtain an explicit rule. As we did a couple of times earlier, using Bayes' rule, we can express $\mathbb{P}(B = \hat{b}|Y = y)$ as:

$$\mathbb{P}(B = \hat{b}|Y = y) = \frac{\mathbb{P}(B = \hat{b}) f_Y(y|B = \hat{b})}{f_Y(y)}. \tag{6}$$

Here we introduce the *density* function $f_Y(y)$ to properly deal with the annoying probability-zero event $Y = y$. Since $f_Y(y)$ is not a function of $B = \hat{b}$, the MAP rule is simplified as:

$$\hat{B}_{\mathsf{MAP}} = \arg \max_{\hat{b}\in\{0,1\}} \mathbb{P}(B = \hat{b}) f_Y(y|B = \hat{b}). \tag{7}$$

Recall the assumption (3): $\mathbb{P}(B=0)=\mathbb{P}(B=1)=\frac{1}{2}$. Hence, under this assumption, it becomes the ML rule:

$$\hat{B}_{\mathsf{MAP}} = \hat{B}_{\mathsf{ML}} = \arg\max_{\hat{b}\in\{0,1\}} \underbrace{f_Y(y|B=\hat{b})}_{\text{likelihood}}. \tag{8}$$

Relationship with the NN rule

Let us prove that the optimal ML rule coincides with our initial guess: the NN rule. We first see that for the Gaussian channel, $f_Y(y|B=1)$ can be rewritten as:

$$\begin{aligned} f_Y(y|B=1) &= f_Y(y|X=+\sqrt{E}) \\ &= f_Z(y-\sqrt{E}|X=+\sqrt{E}) \\ &= f_Z(y-\sqrt{E}) \end{aligned} \tag{9}$$

where the 1st equality comes from the fact that the event $B=1$ is equivalent to the event $X=+\sqrt{E}$ due to our encoding rule PAM; the 2nd is due to the fact that the event $Y=y$ is equivalent to the event $Z=y-X=y-\sqrt{E}$; and the last is because of the independence between Z and X. Here $f_Z(\cdot)=\frac{1}{\sqrt{2\pi\sigma^2}}e^{-\frac{(\cdot)^2}{2\sigma^2}}$. So, the ML rule for the Gaussian channel is:

decide $\hat{B}=1$ if

$$f_Z(y-\sqrt{E}) \geq f_Z(y+\sqrt{E}) \tag{10}$$

and 0 otherwise.

In the equal case $f_Z(y-\sqrt{E})=f_Z(y+\sqrt{E})$, we may want to instead flip a fair coin and to make a random decision. For simplicity, here we take the decision $\hat{B}=1$, which still respects the ML principle.

Using the Gaussian pdf, we can further simplify the above condition (10). It is equivalent to:

$$(y-\sqrt{E})^2 \leq (y+\sqrt{E})^2. \tag{11}$$

Massaging the above a bit, we obtain the further simplified yet equivalent condition:

$$y \geq 0. \tag{12}$$

In summary, the ML decision rule takes the received voltage $Y=y$ and estimates $\hat{B}$ as follows:

$$\begin{aligned} &y \geq 0 && \Longrightarrow \hat{B}=1 \text{ was sent;} \\ &\text{Otherwise} && \Longrightarrow \hat{B}=0 \text{ was sent.} \end{aligned}$$

Fig. 5 illustrates the ML decision rule. The decision rule picks the transmitted voltage level that is closer to the received voltage (closer in the usual sense of Euclidean distance). Hence, the ML rule is exactly the NN rule.

Look ahead

We have derived the optimal MAP receiver when sending one bit via PAM, and found that under the equal chance assumption of B (3), the optimal receiver is equivalent to the ML and NN rule. One natural question that arises is: How about the performance of the optimal receiver? One popular performance measure in the communication context is the *error probability*:

$$P_e := \mathbb{P}(\hat{B}_{\mathsf{MAP}} \neq B) \tag{13}$$

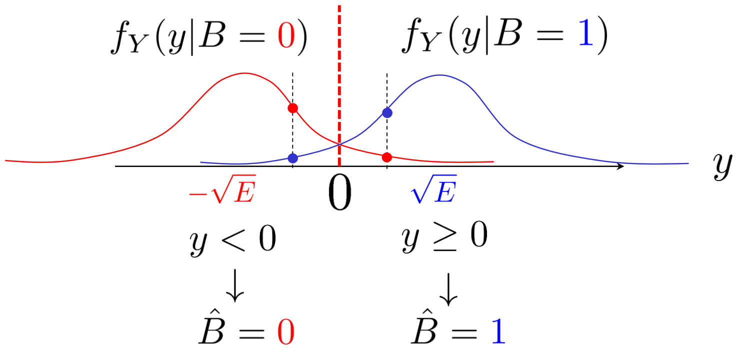


Figure 5: ML rule = NN rule.

In typical communication systems, P_e should be very small for ensuring reliable communication. A good order of such P_e is around $10^{-6} \sim 10^{-10}$. However, one can verify that under the above transmission scheme, P_e is not sufficiently small, i.e., a bit far from the desired range. Don't worry. We will check this next lecture. Then, a follow-up question is: Is there any way to make P_e arbitrarily close to 0 so that we always hit the desired range? It turns out there is another still-simple transmission scheme that does so. Next time, we will investigate the scheme and will demonstrate that P_e can indeed be made arbitrarily small under the scheme.

EE210 Probability and Introductory Random Processes May 13, 2021
KAIST, Spring 2021 Changho Suh (chsuh@kaist.ac.kr)

Lecture 20: Communication: Repetition coding

Recap

Last time we have investigated a simple transmission scheme for sending one bit, and figured out the role of the MAP and ML principles in the design of the optimal receiver under the additive Gaussian noise channel depicted in Fig. 1. Specifically the encoding rule, named PAM, was to map $B = 1$ into $X = +\sqrt{E}$; $B = 0$ to $X = -\sqrt{E}$. In this setup, we demonstrated that the optimal receiver was MAP, and under the reasonable assumption $\mathbb{P}(B = 0) = \mathbb{P}(B = 1) = \frac{1}{2}$, it reduces to the ML rule, and further simplified to the intuitive NN rule.

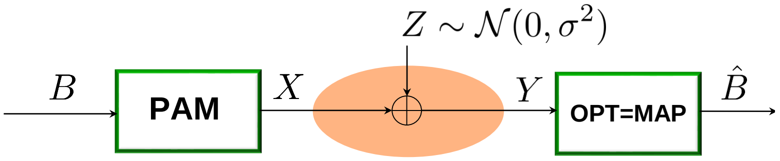


Figure 1: A single-bit transmission via PAM over the additive Gaussian noise channel.

At the end of the last lecture, we raised a question. How about for error probability performance of the optimal receiver?

$$P_e := \mathbb{P}(\hat{B}_{\sf MAP} \neq B). \tag{1}$$

I then claimed that P_e is not sufficiently small, not within a desired range of $10^{-6} \sim 10^{-10}$. But the good news is that there is another yet still-simple way to make P_e arbitrarily close to 0 so that we meet the desired requirement of error probability.

Today's lecture

Today we will explore the way that enables reliable communication. This lecture consists of four parts. First off, we will introduce another communication resource that one can readily think of. That is, *time*. We will then investigate the statistical behaviors of noise signals spread over multiple time slots. Next we will derive the optimal receiver w.r.t. a simple multi-shot transmission scheme that will be discussed soon. Finally, we will analyze the error probability performance of the multi-shot communication scheme equipped with the optimal receiver, thereby demonstrating that P_e can easily be made arbitrarily close to 0.

Time is another communication resource

Recall the previous transmission scheme introduced last time. Here one key observation is that we employ *only one time slot*, although there can be much more time slots available for communication. In other words, we never exploited another very natural communication resource: *time*. Hence, one natural alternative is to employ *multiple time slots*. Actually in Lecture 14, we did the same thing yet in the context of the inference problem for disease testing. *Multiple experiments* are available in testing, so an easy way to boost performance is to employ multiple tests.

Sending one bit over n time slots

We employ a multi-shot transmission scheme. Suppose we send still one bit B but now through n time slots. Assume that the energy budget per time slot is E. A natural question that one can ask is then: What is the best multi-shot communication scheme that yields the minimum error probability given the constraint? In fact, it is a very difficult question to answer. It concerns the central topic of a communication-relevant field named "Information Theory". You know what? Claude Shannon whom I mentioned in Lecture 18 is the Farther of Information Theory. Instead of diving into details for the best communication scheme built upon information theory, here we will take one naive trial based on a very simple idea that people often employ for our daily-life conversation: *repeating what we said*.

So a transmission scheme based on the repetition idea could be: sending a voltage level $\pm\sqrt{E}$ at time 1 and send the *same* voltage again at the other time slots. See Fig. 2. A fancy term for this kind of a scheme is *repetition coding*. For this coding scheme, the received signals read: for $i \in \{1, 2, \ldots, n\}$,

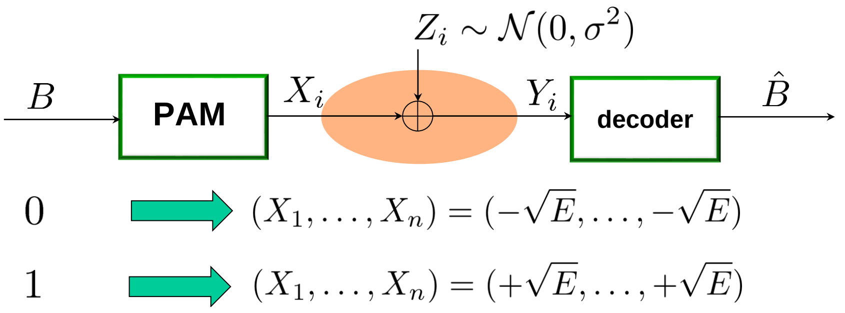

$Z_i \sim \mathcal{N}(0, \sigma^2)$
B
PAM
X_i
Y_i
decoder
$\hat{B}$
0
$(X_1, \ldots, X_n) = (-\sqrt{E}, \ldots, -\sqrt{E})$
1
$(X_1, \ldots, X_n) = (+\sqrt{E}, \ldots, +\sqrt{E})$

Figure 2: Repetition coding: Sending the same voltage signal repeatedly over n time slots.

$$Y_i = X_i + Z_i = \begin{cases} -\sqrt{E} + Z_i, & \text{if } B = 0; \\ +\sqrt{E} + Z_i, & \text{if } B = 1. \end{cases} \tag{2}$$

A probabilistic model for $\{Z_i\}_{i=1}^n$

Our previous discussion in Lecture 18 lets us argue that the statistics of the additive noise at any time slot is *Gaussian*. In practice, the statistics does not change over time significantly. So one reasonable assumption is that the mean and the variance are unchanged over time. Also the noises have little correlation across different time slots. So another reasonable assumption is that Z_i's are mutually *independent*. Simply put, the noises are assumed to be i.i.d. Actually there is a terminology which indicates such a noise model. It is said to be *white*. So we will refer to this noise as the Additive White Gaussian Noise or simply AWGN. You may wonder why we name it "white". It turns out the AWGN contains the frequency components that span the entire spectrum. So it has the same property that the *white light* has: containing every frequency component. That's why people call it white.

Optimal decision rule

Given $(Y_1, \ldots, Y_n) = (y_1, \ldots, y_n)$, the optimal receiver is again MAP (by definition):

$$\hat{B}_{\mathsf{MAP}} = \arg \max_{\hat{b} \in \{0,1\}} \mathbb{P}(B = \hat{b} | Y_1 = y_1, \ldots, Y_n = y_n). \tag{3}$$

As we showed a couple of times earlier, under the assumption $\mathbb{P}(B = 0) = \mathbb{P}(B = 1) = \frac{1}{2}$, the MAP reduces the ML decision rule:

$$\hat{B}_{\mathsf{MAP}} = \hat{B}_{\mathsf{ML}} = \arg \max_{\hat{b} \in \{0,1\}} f_{Y_1,\ldots,Y_n}(y_1, \ldots, y_n | B = \hat{b}). \tag{4}$$

Now consider the likelihood function of interest:

$$\begin{aligned}
& f_{Y_1,\ldots,Y_n}\left(y_1, \ldots, y_n | B = \hat{b}\right) \\
& \overset{(a)}{=} f_{Z_1,\ldots,Z_n}\left(y_1 - X_1, \ldots, y_n - X_n | B = \hat{b}\right) \\
& \overset{(b)}{=} f_{Z_1,\ldots,Z_n}\left(y_1 - (2\hat{b}-1)\sqrt{E}, \ldots, y_n - (2\hat{b}-1)\sqrt{E} | B = \hat{b}\right) \\
& \overset{(c)}{=} f_{Z_1,\ldots,Z_n}\left(y_1 - (2\hat{b}-1)\sqrt{E}, \ldots, y_n - (2\hat{b}-1)\sqrt{E}\right) \\
& \overset{(d)}{=} \prod_{i=1}^{n} f_{Z_i}\left(y_i - (2\hat{b}-1)\sqrt{E}\right) \\
& \overset{(e)}{=} \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2\sigma^2}\left(y_i - (2\hat{b}-1)\sqrt{E}\right)^2\right) \\
& = \left(\frac{1}{\sqrt{2\pi}\sigma}\right)^n \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}\left(y_i^2 + (2\hat{b}-1)^2 E - 2(2\hat{b}-1)\sqrt{E} y_i\right)\right) \\
& \overset{(f)}{=} \left(\frac{1}{\sqrt{2\pi}\sigma}\right)^n \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(y_i^2 + E)\right) \times \exp\left(\frac{1}{\sigma^2}(2\hat{b}-1)\sqrt{E}\sum_{i=1}^{n} y_i\right)
\end{aligned}$$

where (a) follows from the fact that the event $Y_i = y_i$ is equivalent to the event $Z_i = y_i - X_i$; (b) is because the simplest expression of our encoding rule reads $X_i = (2B-1)\sqrt{E}$; (c) is due to the independence between $\{Z_i\}_{i=1}^n$ and B; (d) is due to the independence of the additive noises at different time slots; (e) comes from the pdf of the Gaussian noise; and (f) comes from the fact that $(2\hat{b}-1)^2 = 1$ always because $\hat{b} \in \{0, 1\}$.

Here one key observation is that the first part in the last equality (marked in red) is *irrelevant* of $\hat{b}$. Hence, by taking a logarithmic function (an increasing function) w.r.t. the remaining relevant part, we get:

$$\begin{aligned}
\hat{B}_{\mathsf{ML}} &= \arg \max_{\hat{b} \in \{0,1\}} f_{Y_1,\ldots,Y_n}(y_1, \ldots, y_n | B = \hat{b}) \\
&= \arg \max_{\hat{b} \in \{0,1\}} \exp\left(\frac{(2\hat{b}-1)\sqrt{E}}{\sigma^2}\sum_{i=1}^{n} y_i\right) \\
&= \arg \max_{\hat{b} \in \{0,1\}} (2\hat{b}-1)\sum_{i=1}^{n} y_i.
\end{aligned} \tag{5}$$

Notice that the sum $\sum_{i=1}^n y_i$ plays a significant role in the decision:

$\sum_{i=1}^n y_i \geq 0 \Longrightarrow \hat{B}_{\mathsf{ML}} = 1$;

$\sum_{i=1}^n y_i < 0 \Longrightarrow \hat{B}_{\mathsf{ML}} = 0$.

Again we declare $\hat{B}_{\mathsf{ML}} = 1$ when the equality occurs.

What sense can we make of this rule? We are collecting all the received signals and taking some sort of a "joint opinion". If the sum is positive, we decide that the bit must correspond to a positive signal (and vice-versa). One can interpret this as a kind of majority voting, more precisely, a soft version of majority voting. Why? Here what I mean by majority voting is that: we first make a hard decision for each received signal ($y_i \geq 0 \longrightarrow \hat{B}_i = 1$; otherwise, $\hat{B}_i = 0$); and then declares the one that receives the most votes ($\sum_{i=1}^{n} \hat{B}_i \geq \frac{n}{2} \longrightarrow \hat{B}_{\mathsf{voting}} = 1$). But the ML rule relies upon aggregation of y_i's and then makes a decision. So only one decision is made w.r.t sort of a soft version of all the opinions. This rule coincides with our natural guess: the NN decision rule yet w.r.t. the sum $\sum_{i=1}^{n} Y_i$.

Error probability

Now let us analyze the error probability:

$$P_e = \mathbb{P}(\hat{B}_{\mathsf{ML}} \neq B). \tag{6}$$

Two random quantities appear in the interested error event: B and $\hat{B}$. How to deal with this? Yes, we should rely upon the *total probability law*. Using this, we obtain:

$$\begin{aligned}\mathbb{P}(\hat{B}_{\mathsf{ML}} \neq B) &= \mathbb{P}(B=0, \hat{B}_{\mathsf{ML}} \neq B) + \mathbb{P}(B=1, \hat{B}_{\mathsf{ML}} \neq B) \\ &= \mathbb{P}(B=0)\mathbb{P}(\hat{B}_{\mathsf{ML}} = 1 | B=0) + \mathbb{P}(B=1)\mathbb{P}(\hat{B}_{\mathsf{ML}} = 0 | B=1)\end{aligned} \tag{7}$$

where the 2nd equality comes from the definition of conditional probability. As mentioned earlier, we assume that the a priori probabilities are equal (to 0.5 each). Let us focus on one of the error events by assuming that the information bit was actually 0. Then with the NN rule, we get:

$$\begin{aligned}\mathbb{P}(\hat{B}_{\mathsf{ML}} = 1 | B=0) &\overset{(a)}{=} \mathbb{P}\left(\sum_{i=1}^{n} Y_i \geq 0 | B=0\right) \\ &= \mathbb{P}\left(-n\sqrt{E} + \sum_{i=1}^{n} Z_i \geq 0 | B=0\right) \\ &\overset{(b)}{=} \mathbb{P}\left(\sum_{i=1}^{n} Z_i \geq n\sqrt{E}\right) \\ &= \mathbb{P}\left(\frac{\sum_{i=1}^{n} Z_i}{\sqrt{n}\sigma} \geq \frac{\sqrt{nE}}{\sigma}\right) \\ &\overset{(c)}{=} \int_{\frac{\sqrt{nE}}{\sigma}}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{z^2}{2}} dz.\end{aligned} \tag{8}$$

where (a) is due to the NN rule[1]; (b) is because of the independence between $\{Z_i\}_{i=1}^{n}$ and B; and (c) follows from the fact that $\frac{\sum_{i=1}^{n} Z_i}{\sqrt{n}\sigma} \sim \mathcal{N}(0,1)$ (Why? Think about it). The quantity in the last equation in the above (8) indicates the area of a right tail part of the normal Gaussian pdf. Actually it often appears in the performance analysis of many communication schemes. Hence, there is a terminology indicating the term. It is called the *Q-function* and denoted by

$$Q(z) := \int_{z}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt. \tag{9}$$

[1] For simplicity of analysis, we assume that for the event $Y = 0$, $\hat{B}$ is decided to be 1. Since it is the probability-zero event, the error probability analysis remains the same.

Using this, we then get:

$$\mathbb{P}(\hat{B}_{\sf ML} = 1|B = 0) = Q\left(\frac{\sqrt{nE}}{\sigma}\right). \tag{10}$$

Here the Q-function is always tabulated in all probability textbooks and wikipedia. This integration can also be computed numerically in Python.

Consider $\mathbb{P}(\hat{B}_{\sf ML} = 0|B = 1)$. Actually the error does not depend on which information bit is transmitted. The complete *symmetry* of the mapping from the bit values to the voltage levels and the NN decision rule would suggest that the two error probabilities are identical. For completeness, we go through the calculation for $\mathbb{P}(\hat{B}_{\sf ML} = 0|B = 1)$ and verify that it is indeed the same:

$$\begin{aligned}\mathbb{P}(\hat{B}_{\sf ML} = 0|B = 1) &= \mathbb{P}\left(\sum_{i=1}^{n} Y_i < 0|B = 1\right)\\ &= \mathbb{P}\left(n\sqrt{E} + \sum_{i=1}^{n} Z_i < 0|B = 1\right)\\ &= \mathbb{P}\left(\frac{\sum_{i=1}^{n} Z_i}{\sqrt{n}\sigma} < -\frac{\sqrt{nE}}{\sigma}\right)\\ &= Q\left(\frac{\sqrt{nE}}{\sigma}\right)\end{aligned} \tag{11}$$

where the last equality is due to the symmetry of the Gaussian pdf. Applying this (together with (10)) into (7), we obtain:

$$P_e = Q\left(\frac{\sqrt{nE}}{\sigma}\right). \tag{12}$$

Error probability vs. n

We are now ready to discuss the error probability as a function of n. Prior to this, let me emphasize one important measure that often appears in communication. That is, the ratio between the energy budget E and the noise variance σ^2:

$$\mathsf{SNR} := \frac{E}{\sigma^2}. \tag{13}$$

The ratio has a famous name: SNR. It stands for the Signal-to-Noise energy Ratio. It acts as an intuitive measure that reflects the goodness of an interested channel: the larger, the better. A typical value of SNR in many of the practically-relevant channels: $0 \sim 20$ dB. It is common to use the dB scale for SNR: SNR dB $= 10\log_{10}\mathsf{SNR}$. So the typical range is translated into $\mathsf{SNR} = 1 \sim 100$.

To get a concrete feel as to how P_e behaves as a function of n, we plot a logarithmic-scaled curve of P_e as a function of n, for a typical value of $\mathsf{SNR} = 10$ dB. See Fig. 3. We see that P_e decays *exponentially* in n, thus hitting the desired range of $P_e = 10^{-6} \sim 10^{-10}$ with only a few time slots.

Look ahead

During the past three lectures, we demonstrated the role of probabilistic modeling, MAP and ML principles for communication. Next time, we will move onto the 2nd application: Machine learning.

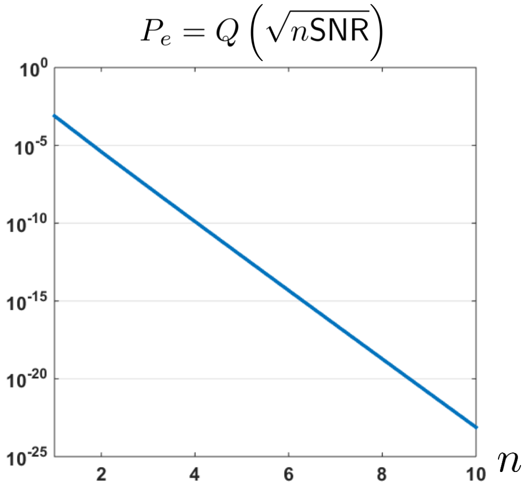


Figure 3: P_e vs n: $\log_{10} P_e$ as a function of n.

EE210 Probability and Introductory Random Processes May 18, 2021
KAIST, Spring 2021 Changho Suh (chsuh@kaist.ac.kr)

Lecture 21: Machine learning: Probabilistic modeling

Recap

During the past three lectures, we have explored the role of probability in communication. We first focused on an uncertain entity that arises in communication: *noise*. Translating the physical properties of the noise into a mathematical framework and then applying the central limit theorem, we then showed that the noise can be modeled as a *Gaussian* random variable. Next we demonstrated the role of the MAP and ML principles in the design of the optimal receiver w.r.t. the additive Gaussian noise channel. Finally we analyzed the performance of the optimal receiver via a probabilistic measure: *probability of error*.

Today's lecture

Today we will embark on the 2nd application: *Machine learning*. This lecture consists of four parts. We will first review the definition of machine learning that we briefly looked through in Lecture 1. We will then discuss how machine learning is related to probability, identifying a *probabilistic* aspect in machine learning. Next we will formulate an optimization problem that plays a crucial role in the design of a machine learning model. Lastly we will investigate one function that arises in the optimization formulation, and will emphasize that the choice of the function is closely related to the ML principle.

Review: Machine learning

Let us start by reviewing the definition of machine learning. Machine learning is a methodology for training a machine so that it can perform like human beings. Formally speaking, machine learning is the study of *algorithms* (a set of instructions that computer can execute) with which one can train a computer system so that the trained machine can perform a specific task of interest. Pictorially, it means the following; see Fig. 1.

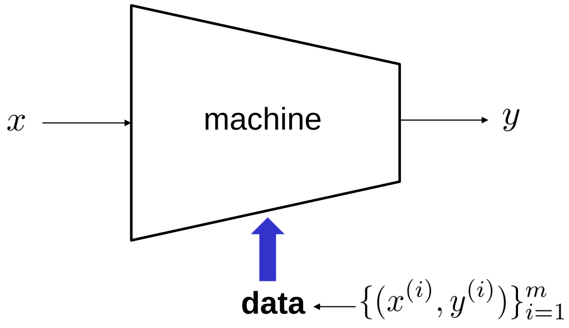


Figure 1: Machine learning: A methodology for training a machine based on *data* so that it can perform like human beings.

Here the entity that we are interested in building up is a computer system, which is definitely a *machine*. Since it is a system (i.e., a function), it has an input and an output. The input, usually denoted by x, indicates information which is employed to perform a task of interest.

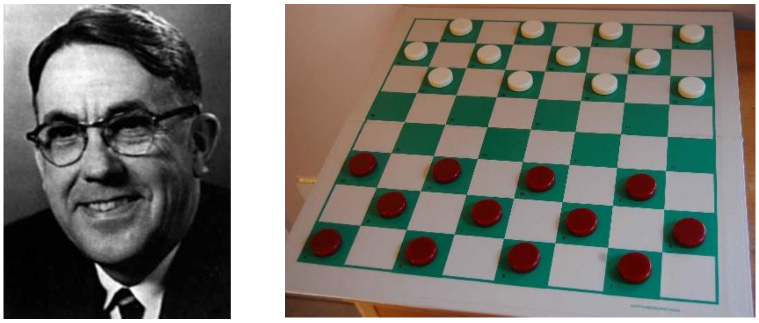
The output, usually denoted by y, indicates a task result. For instance, if a task of interest is cat-vs-dog image classification, x could be *image-pixel values* and y is a binary value indicating whether the fed image is a cat (say $y = 1$) or a dog ($y = 0$). One crucial aspect of machine learning is that we use *data* in the process of training a machine. The data often refers to input-output paired samples, denoted by:

$$\{(x^{(i)}, y^{(i)})\}_{i=1}^{m}, \tag{1}$$

where $(x^{(i)}, y^{(i)})$ indicates the ith input-output sample and m denotes the number of samples. Remember people often use a different terminology, an *example*, to indicate a sample.

A remark on the naming

One can easily see the rationale of the naming via changing a viewpoint. From a *machine's perspective*, a *machine learns* the task from data. Hence, it is called *machine learning.* This naming was coined in 1959 by Arthur Lee Samuel. See Fig. 2.



Arthur Samuel '59

checkers

Figure 2: Arthur Lee Samuel is an American pioneer in the field of artificial intelligence. One of his prominent achievements in early days is to develop computer checkers which later formed the basis of AlphaGo.

Arthur Samuel is actually one of the pioneers in *Artificial Intelligence* (AI) which includes machine learning as a sub-field. The AI field is the study of creating *intelligence* by *machines*, unlike the *natural intelligence* displayed by intelligent beings like humans and animals.

One of his achievements in early days is to develop a human-like computer player for a board game, called *checkers*; see the right figure in Fig. 2. He proposed many algorithms and ideas while developing computer checkers. It turns out those algorithms could form the basis of *AlphaGo*, a computer program for the board game Go which defeated one of the 9-dan professional players, Lee Sedol, with 4 wins out of 5 games in 2016.

A probabilistic aspect in machine learning

The relationship between machine learning and probability is via data $\{(x^{(i)}, y^{(i)})\}_{i=1}^{m}$. This is because people often view the data as one particular realization of a *random process*:

$$\{(X^{(i)}, Y^{(i)})\}_{i=1}^{m}. \tag{2}$$

One natural assumption that we often made in reality is that the random process is i.i.d. across distinct examples, each being distributed according to a joint distribution $\mathbb{P}_{X,Y}(x, y)$:

$$\{(X^{(i)}, Y^{(i)})\}_{i=1}^{m} \text{ i.i.d. } \sim \mathbb{P}_{X,Y}(x, y). \tag{3}$$

Training via optimization

Another probabilistic aspect in machine learning is related to a training methodology. In order to understand what this means, let us first discuss a common way to train a machine, i.e., estimate a function of machine, say $f(\cdot)$. The common way to estimate $f(\cdot)$ is through solving an *optimization problem*. You may then wonder how an optimization problem is related to training a machine?

Objective function

To figure this out, let us consider what could be an *objective function* in the machine learning task. What we want in the design of a machine learning model is that the ground-truth label $y^{(i)}$ is close to the prediction $f(x^{(i)})$ as much as possible for all examples:

$$y^{(i)} \approx f(x^{(i)}), \ \ \forall i \in \{1, \ldots, m\}.$$

A natural question that arises is then: How to quantify *closeness* (reflected in the "≈" notation) between the two quantities: $y^{(i)}$ and $f(x^{(i)})$? One very common way that has been used in the field is to employ a function, called a *loss* function, usually denoted by:

$$\ell(y^{(i)}, f(x^{(i)})). \tag{4}$$

One obvious property that the loss function $\ell(\cdot, \cdot)$ should have is that it should be small when the two arguments are close, while being zero when the two are identical. Using such loss function (4), one can then formulate an optimization problem as:

$$\min_{f(\cdot)} \sum_{i=1}^{m} \ell(y^{(i)}, f(x^{(i)})). \tag{5}$$

How to introduce optimization variable?

Next, what is the best way that minimizes the objective function? To figure this out, we first need to identify a quantity, so called the optimization variable, which affects the objective function. Unfortunately, there is no variable. Instead we have a different quantity that we can optimize over: *the function* $f(\cdot)$, marked in red in (5). The question is then: How to introduce optimization variable? A common way employed in the field is to represent the function $f(\cdot)$ with *parameters (or called weights)*, denoted by w, and then consider such weights as an optimization variable. Taking this approach, one can then translate the problem (5) into:

$$\min_{w} \sum_{i=1}^{m} \ell(y^{(i)}, f_{w}(x^{(i)})) \tag{6}$$

where $f_w(x^{(i)})$ denotes the function $f(x^{(i)})$ parameterized by w.

The above optimization problem depends on how we define the two functions: (i) $f_w(x^{(i)})$ w.r.t. w; (ii) the loss function $\ell(\cdot, \cdot)$. In machine learning, lots of works have been done for the choice of such functions.

A choice for $f_w(\cdot)$

Around at the same time when the machine learning field was founded, one architecture was suggested for the function $f_w(\cdot)$ in the context of simple binary classifiers in which y takes one among the two options only, e.g., $y \in \{0, 1\}$. The architecture is called:

$$Perceptron,$$

and was invented in 1957 by one of the pioneers in the AI field, named Frank Rosenblatt. See Fig. 3. Interestingly, Frank Rosenblatt was a *psychologist*. He was interested in how brains of intelligent beings work and his study on brains led him to come up with the perceptron, and therefore gave significant insights into *neural networks* that many of you guys heard of.

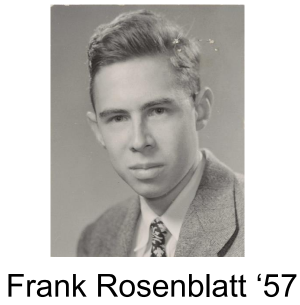


Figure 3: Frank Rosenblatt (1928–1971) is an American psychologist notable as the inventor of perceptron. One sad story is that he died in 1971 on his 43rd birthday, in a boating accident.

How brains work

Here are details on how the brain structure inspired the perceptron architecture. Inside a brain, there are many *electrically excitable cells*, called *neurons*; see Fig. 4. Here a red-circled one indicates a neuron. So the figure shows three neurons in total. There are three major properties about neurons that led to the architecture.

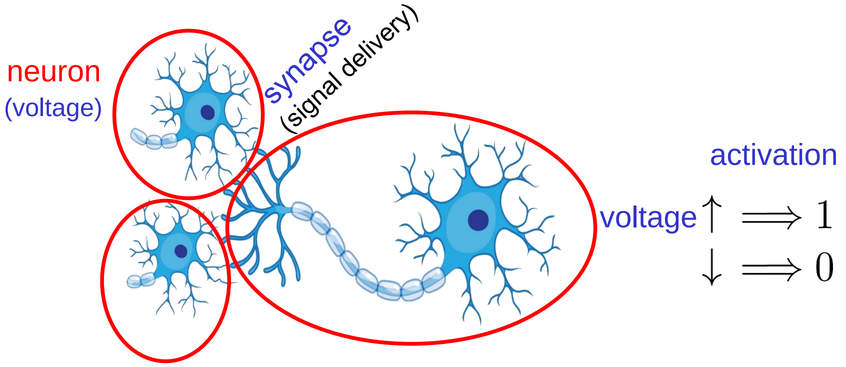


Figure 4: Neurons are electrically excitable cells and are connected through synapses.

The first is that a neuron is an *electrical* quantity, so it has a *voltage*. The second property is that neurons are connected with each other through mediums, called *synapses*. So the main role of synapses is to deliver electrical voltage signals across neurons. Depending on the connectivity strength level of a synapse, a voltage signal from one neuron to another can increase or decrease. The last is that a neuron takes a particular action, called *activation*. Depending on its voltage level, it generates an all-or-nothing pulse signal. For instance, if its voltage level is above a certain threshold, then it generates an impulse signal with a certain magnitude, say 1; otherwise, it produces nothing.

Perceptron

The above three properties led Frank Rosenblatt to propose the perceptron architecture, as illustrated in Fig. 5.

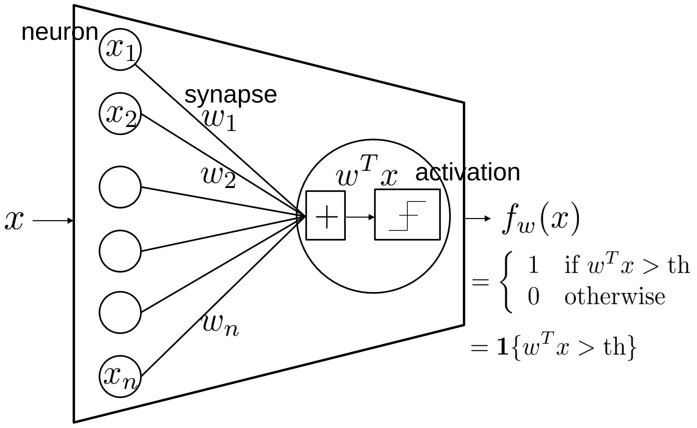

neuron
x_1
x_2
synapse
w_1
w_2
w_n
x_n
x
$w^T x$
activation
+
$f_w(x)$
$= \begin{cases} 1 & \text{if } w^T x > \text{th} \\ 0 & \text{otherwise} \end{cases}$
$= \mathbf{1}\{w^T x > \text{th}\}$

Figure 5: The architecture of perceptron.

Let x be an n-dimensional real-valued signal: $x := [x_1, x_2, \ldots, x_n]^T$. Suppose each component x_i is distributed to each neuron, and let x_i indicate a voltage signal level of the ith neuron. The voltage signal x_i is then delivered through a synapse to another neuron (placed on the right in the figure, indicated by a big circle). Remember that the voltage level can increase or decrease depending on the connectivity strength of a synapse. To capture this, a weight, say w_i, is multiplied to x_i so $w_i x_i$ is a delivered voltage signal at the terminal neuron. Based on an empirical observation that the voltage level at the terminal neuron increases with more connected neurons, Rosenblatt introduced an adder which simply aggregates all the voltage signals coming from many neurons, so he modeled the voltage signal at the terminal neuron as:

$$w_1 x_1 + w_2 x_2 + \cdots + w_n x_n = w^T x. \tag{7}$$

Lastly in an effort to mimic the *activation*, he modeled the output signal as

$$f_w(x) = \begin{cases} 1 & \text{if } w^T x > \text{th}; \\ 0 & \text{otherwise,} \end{cases} \tag{8}$$

where "th" indicates a certain threshold level. It can also be simply denoted as

$$f_w(x) = \mathbf{1}\{w^T x > \text{th}\}. \tag{9}$$

Activation functions

Taking the percentron as a function class, one can formulate the optimization problem (6) as:

$$\min_w \sum_{i=1}^{m} \ell(y^{(i)}, \mathbf{1}\{w^T x^{(i)} > \text{th}\}). \tag{10}$$

This is an initial optimization problem that people came up with. However, people figured out there is an issue in solving this optimization. The issue comes from the fact that the objective function contains an indicator function, so it is *not differentiable*. As we saw a couple of times earlier, a common way to solve an optimization problem includes "derivative computation". Hence, the above optimization (10) including the non-differentiable function is not desirable. What can we do then? One typical way that people have taken in the field is to *approximate* the activation function. There are many ways for approximation. From below, we will investigate one of them.

Approximate the original step-function activation!

One popular way is to use the following function that makes a *smooth* transition from 0 to 1:

$$f_w(x) = \frac{1}{1+e^{-w^T x}}. \tag{11}$$

Notice that $f_w(x) \approx 0$ when $w^T x$ is very small; it then grows exponentially with an increase in $w^T x$; later grows logarithmically; and finally saturates as 1 when $w^T x$ is very large. See Fig. 6. Actually the function (11) is a very popular one used in statistics, called the *logistic*[1] function.

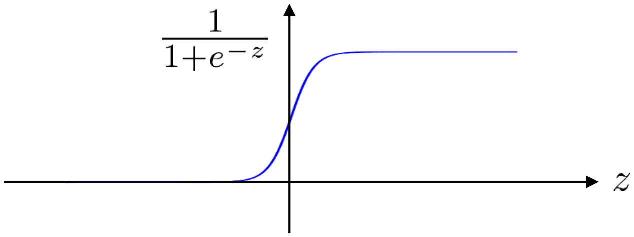


Figure 6: Logistic function: $\sigma(z) = \frac{1}{1+e^{-z}}$.

There is another name for the function, which is the *sigmoid*[2] function.

There are two good features about the logistic function. First it is differentiable. The second is that it can be interpreted as the *probability* for the output in the binary classifier, e.g., $\mathbb{P}(Y = 1)$ where Y denotes a random variable for the ground-truth label in the binary classifier. So it is interpretable.

Look ahead

Under the choice of the logistic activation, what is a good choice for a *loss* function? It turns out that the ML principle plays an important role in the design of an *optimal* loss function in some sense. Next time we will investigate in what sense it is optimal. We will then figure out how the principle comes up in the design of the optimal loss function.

[1] The word *logistic* comes from a Greek word which means a slow growth, like a logarithmic growth.

[2] Sigmoid means resembling the lower-case Greek letter sigma, S-shaped.

EE210 Probability and Introductory Random Processes — May 20, 2021
KAIST, Spring 2021 — Changho Suh (chsuh@kaist.ac.kr)

Lecture 22: Machine learning: ML principle

Recap

Last time we formulated an optimization problem for the design of a machine learning model based on the perceptron architecture:

$$\min_w \sum_{i=1}^m \ell(y^{(i)}, \hat{y}^{(i)}) \tag{1}$$

where $\{(x^{(i)}, y^{(i)})\}_{i=1}^m$ indicate input-output paired examples; $\ell(\cdot,\cdot)$ denotes a loss function; and $\hat{y}^{(i)} := f_w(x^{(i)})$ is the prediction parameterized by the weights w. As an activation function, we introduced a logistic function that is widely used in the field:

$$f_w(x) = \frac{1}{1+e^{-w^T x}}. \tag{2}$$

At the end of the last lecture, I claimed that the ML principle plays a crucial role in the design of the *optimal* loss function.

Today's lecture

Today we will discuss details on the claim. Specifically what we are going to do are three-folded. We will first investigate what it means by the *optimal* loss function. We will next figure out how the ML principle comes up in the design of the optimal loss function. Lastly we will do one more thing. That is, studying how to solve the formulated optimization. It turns out there is no closed-form solution to the optimization, but instead there is a prominent algorithm which enables us to obtain a *numerical* solution with the help of a computer. The algorithm is called *gradient descent*. So in the last part, we will figure out gradient descent.

Optimality in a sense of maximizing likelihood

A binary classifier with the logistic function (2) is called *logistic regression*. Actually this naming is a bit confusing, as "regression" means prediction, not classification. The use of such naming is because the classifier yields a continuous value for prediction, instead of a discrete value, say between 0 and 1. See Fig. 1 for illustration.

Notice that the output $\hat{y}$ lies in between 0 and 1:

$$0 \le \hat{y} \le 1.$$

Hence, one can interpret the output as a *probabilistic* quantity. One natural assumption that we can make inspired by the interpretation is:

$$\text{Assumption} : \hat{y} = \mathbb{P}(Y=1|X=x) \tag{3}$$

where X and Y denote random variables for input and output, respectively. This makes an intuitive sense. A large value of $\hat{y}$ (close to 1) leads us to naturally decide the ground-truth as 1, while a small value of $\hat{y}$ (close to 0) leads to the ground-truth 0.

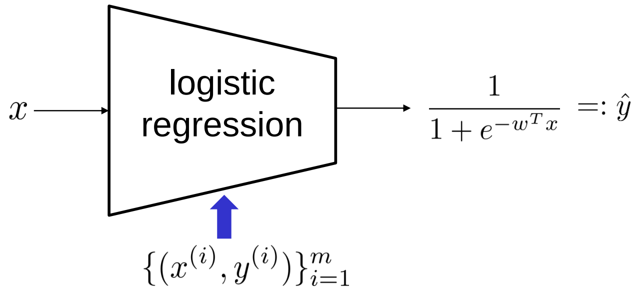

x
logistic regression
$\frac{1}{1+e^{-w^T x}} =: \hat{y}$
$\{(x^{(i)}, y^{(i)})\}_{i=1}^m$

Figure 1: Logistic regression

The meaning of the optimal solution to logistic regression can be defined under the above assumption. In order to figure out what this means, first consider the *likelihood* of the ground-truth classifier:

$$\mathbb{P}\left(Y^{(1)} = y^{(1)}, \ldots, Y^{(m)} = y^{(m)} | X^{(1)} = x^{(1)}, \ldots, X^{(m)} = x^{(m)}\right). \tag{4}$$

Notice that the classifier output $\hat{y}$ is a function of weights w, as the classifier is parameterized by w. Hence, assuming (3), the likelihood (4) is also a function of w.

We are now ready to define the optimality of w. The optimal weight, say w^*, is defined as the one that *maximizes likelihood* (4):

$$w^* := \arg\max_w \mathbb{P}\left(Y^{(1)} = y^{(1)}, \ldots, Y^{(m)} = y^{(m)} | X^{(1)} = x^{(1)}, \ldots, X^{(m)} = x^{(m)}\right). \tag{5}$$

Of course, there are other ways to define the optimality. Here, we employ the ML principle, the most popular choice. This is exactly where the definition of the *optimal loss function*, say $\ell^*(\cdot, \cdot)$, kicks in. We say that $\ell^*(\cdot, \cdot)$ is defined as the one that satisfies:

$$\arg\min_w \sum_{i=1}^m \ell^*(y^{(i)}, \hat{y}^{(i)}) = \arg\max_w \mathbb{P}\left(Y^{(1)} = y^{(1)}, \ldots, Y^{(m)} = y^{(m)} | X^{(1)} = x^{(1)}, \ldots, X^{(m)} = x^{(m)}\right). \tag{6}$$

It turns out the condition (6) would give us the optimal loss function $\ell^*(\cdot, \cdot)$ that yields a very well-known machine learning classifier: *logistic regression*, in which the loss function reads:

$$\ell^*(y, \hat{y}) = \ell_{\textsf{logistic}}(y, \hat{y}) = -y \log \hat{y} - (1 - y) \log(1 - \hat{y}). \tag{7}$$

From this, we see that the ML principle plays a key role in the design of the well-known classifier, logistic regression, which is the optimal classifier under the perception architecture. In the next section, we will prove (7).

Derivation of the optimal loss function $\ell^*(\cdot, \cdot)$

Usually samples are obtained from different contexts. Hence, it is reasonable to assume that such samples are independent with each other:

$$\{(X^{(i)}, Y^{(i)})\}_{i=1}^m \text{ are independent over } i. \tag{8}$$

Under this assumption, we can then rewrite the likelihood (4) as:

$$\begin{aligned}
&\mathbb{P}\left(Y^{(1)}=y^{(1)},\ldots,Y^{(m)}=y^{(m)}|X^{(1)}=x^{(1)},\ldots,X^{(m)}=x^{(m)}\right)\\
&\overset{(a)}{=}\frac{\mathbb{P}\left(X^{(1)}=x^{(1)},Y^{(1)}=y^{(1)},\ldots,X^{(m)}=x^{(m)},Y^{(m)}=y^{(m)}\right)}{\mathbb{P}\left(X^{(1)}=x^{(1)},\ldots,X^{(m)}=x^{(m)}\right)}\\
&\overset{(b)}{=}\frac{\prod_{i=1}^{m}\mathbb{P}_{X,Y}\left(x^{(i)},y^{(i)}\right)}{\prod_{i=1}^{m}\mathbb{P}_{X}(x^{(i)})}\\
&\overset{(c)}{=}\prod_{i=1}^{m}\mathbb{P}_{Y|X}\left(y^{(i)}|x^{(i)}\right)
\end{aligned}\tag{9}$$

where (a) and (c) are due to the definition of conditional probability; and (b) comes from the independence assumption (8). Some of you guys may wonder why $\{X_i\}_{i=1}^m$ are also independent. We can readily prove this from (8). Please check if you are not convinced. Here $\mathbb{P}_{X,Y}(x^{(i)},y^{(i)})$ denotes the probability distribution of the input-output pair of the system:

$$\mathbb{P}_{X,Y}(x^{(i)},y^{(i)}):=\mathbb{P}(X=x^{(i)},Y=y^{(i)}).\tag{10}$$

Similarly

$$\mathbb{P}_{X}(x^{(i)}):=\mathbb{P}(X=x^{(i)}).\tag{11}$$

Recall the probability-interpretation-related assumption (3) made with regard to $\hat{y}$:

$$\hat{y}=\mathbb{P}(Y=1|X=x).$$

This implies that:

$$\begin{aligned}
y=1:&\quad \mathbb{P}_{Y|X}(y|x)=\hat{y};\\
y=0:&\quad \mathbb{P}_{Y|X}(y|x)=1-\hat{y}.
\end{aligned}$$

Hence, a succinct representation for $\mathbb{P}_{Y|X}(y|x)$ reads:

$$\mathbb{P}_{Y|X}(y|x)=\hat{y}^{y}(1-\hat{y})^{1-y}.$$

Using the notations of $(x^{(i)},y^{(i)})$ and $\hat{y}^{(i)}$, we then get:

$$\mathbb{P}_{Y|X}\left(y^{(i)}|x^{(i)}\right)=(\hat{y}^{(i)})^{y^{(i)}}(1-\hat{y}^{(i)})^{1-y^{(i)}}.$$

Plugging this into (9), we get:

$$\mathbb{P}\left(Y^{(1)}=y^{(1)},\ldots,Y^{(m)}=y^{(m)}|X^{(1)}=x^{(1)},\ldots,X^{(m)}=x^{(m)}\right)=\prod_{i=1}^{m}(\hat{y}^{(i)})^{y^{(i)}}(1-\hat{y}^{(i)})^{1-y^{(i)}}.\tag{12}$$

Applying this into (5), we obtain:

$$\begin{aligned}
w^{*}&=\arg\max_{w}\prod_{i=1}^{m}(\hat{y}^{(i)})^{y^{(i)}}(1-\hat{y}^{(i)})^{1-y^{(i)}}\\
&\overset{(a)}{=}\arg\max_{w}\sum_{i=1}^{m}y^{(i)}\log\hat{y}^{(i)}+(1-y^{(i)})\log(1-\hat{y}^{(i)})\\
&\overset{(b)}{=}\arg\min_{w}\sum_{i=1}^{m}-y^{(i)}\log\hat{y}^{(i)}-(1-y^{(i)})\log(1-\hat{y}^{(i)})
\end{aligned}\tag{13}$$

where (a) comes from the fact that $\log(\cdot)$ is a non-decreasing function and $\prod_{i=1}^{m}(\hat{y}^{(i)})^{y^{(i)}}(1-\hat{y}^{(i)})^{1-y^{(i)}}$ is positive; and (b) is due to changing the sign of the objective function.

In fact, the term inside the summation in the last equality in (13) respects the formula of an important notion that arises in the field of information theory: *cross entropy.* In particular, in the context of a loss function, it is named *cross entropy loss*:

$$\ell_{\mathsf{CE}}(y,\hat{y}) := -y\log\hat{y} - (1-y)\log(1-\hat{y}). \tag{14}$$

Hence, the optimal loss function that maximizes likelihood is cross entropy loss:

$$\ell^*(\cdot,\cdot) = \ell_{\mathsf{CE}}(\cdot,\cdot).$$

Remarks on cross entropy loss (14)

Let me say a few words about the rationale behind the naming of cross entropy loss (14). Actually this comes from the definition of *cross entropy.* The cross entropy is defined w.r.t. two random variables. For simplicity, let us consider two binary random variables, say $X \sim \mathsf{Bern}(p)$ and $Y \sim \mathsf{Bern}(q)$. For such two random variables, cross entropy is defined as:

$$H(p,q) := -p\log q - (1-p)\log(1-q). \tag{15}$$

Notice that the formula of (15) is exactly the same as the term inside summation in (13), except for having different notations. Hence, it is called *cross entropy loss.* Some curious students may still wonder why the formula (15) is called "cross entropy". In PS7, you will have a chance to know about the rationale.

How to solve (13)?

From (13) and (2), we can write the optimization problem as:

$$\min_{w}\sum_{i=1}^{m} -y^{(i)}\log\frac{1}{1+e^{-w^Tx^{(i)}}} - (1-y^{(i)})\log\frac{e^{-w^Tx^{(i)}}}{1+e^{-w^Tx^{(i)}}}. \tag{16}$$

Let $J(w)$ be the normalized version of the objective function:

$$J(w) := \frac{1}{m}\sum_{i=1}^{m} -y^{(i)}\log\hat{y}^{(i)} - (1-y^{(i)})\log(1-\hat{y}^{(i)}). \tag{17}$$

It turns out the above optimization belongs to *convex optimization* which I briefly mentioned about in Lecture 16. Let me say a few words about convex optimization which can give insights into how to solve the problem. Simply put, convex optimization means a class of optimization problems which can be efficiently solved on a computer. More formally, it is a class of problems wherein the objective is a *convex function* in the minimization problem. Roughly speaking, here the convex function is defined as a bowl-shaped function as illustrated in Fig. 2. This figure represents a single-dimensional case of w for illustrative purpose. For a multi-dimensional case $w := [w_1,\ldots,w_n]$, it means the bowl-shape w.r.t. every w_i where $i \in \{1,\ldots,n\}$. See Appendix #1 for the formal definition; and Appendix #2 for the proof of convexity of $J(w)$. Don't worry. Appendices #1 and #2 are not included in the final exam.

One crucial fact about the convex function (the bowl-shaped function) is that the *minimum* point occurs at the *unique stationary* point, as long as the minimum is finite. We will not prove this, but it makes an intuitive sense from the bowl-shaped curve in Fig. 2. Using this fact, we can then say that w^* is the *stationary* point:

$$\nabla J(w^*) = 0. \tag{18}$$

So the only thing that matters is to figure out such w^*. But there is an issue in deriving w^*. The issue is that analytically finding such point is not doable because it turns out there is no closed-form solution (check!). However, there is a good news. The good news is that there are several algorithms which allow us to find such point efficiently without the knowledge of the closed-form solution. One prominent algorithm that has been widely used in the field is: *gradient descent.*

Gradient descent

Here is how it works. It is an *iterative* algorithm. Suppose that at the t-th iteration, we have an estimate for w^*, say $w^{(t)}$. Usually $w^{(0)}$ is chosen at random. We then compute the gradient of the function evaluated at the estimate: $\nabla J(w^{(t)})$. Next we update the estimate along a direction being *opposite* to the direction of the gradient:

$$w^{(t+1)} \longleftarrow w^{(t)} - \alpha \nabla J(w^{(t)}) \tag{19}$$

where $\alpha > 0$ indicates the stepsize (or called the learning rate). If you think about it, this update rule makes sense. Suppose $w^{(t)}$ is placed right relative to the optimal point w^*, as illustrated in Fig. 2.

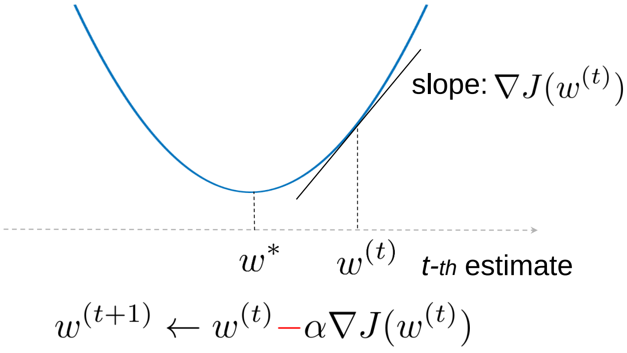

slope: $\nabla J(w^{(t)})$
w^*
$w^{(t)}$
t-th estimate
$w^{(t+1)} \leftarrow w^{(t)} - \alpha \nabla J(w^{(t)})$

Figure 2: Gradient descent.

Then, we should move $w^{(t)}$ to the *left* so that it is closer to w^*. The update rule actually does this, as we *subtract* by $\alpha\nabla J(w^{(t)})$. Notice that $\nabla J(w^{(t)})$ points to the *right* direction given that $w^{(t)}$ is placed right relative to w^*. We repeat this procedure until it converges. It turns out: as $t \rightarrow \infty$, it actually converges:

$$w^{(t)} \longrightarrow w^*, \tag{20}$$

as long as the learning rate is chosen properly, like the one delaying exponentially w.r.t. t, e.g., e^{-t}. We will not touch upon the proof of this convergence. Actually the proof is not that simple – even there is a big field in statistics which intends to prove the convergence of a variety of algorithms (if it is the case).

Look ahead

We have so far investigated the role of probabilistic modeling and the MAP/ML principles for two applications: communication and machine learning. For a couple of upcoming lectures, we will do the same thing for the last application: speech recognition. You will figure out that speech recognition is a beautiful application which concerns almost all the important concepts and principle covered so far.

Appendix #1: Convex functions

An informal yet intuitive definition of a convex function is the following. We say that a function is convex if it is bowl-shaped, as illustrated in Fig. 3.

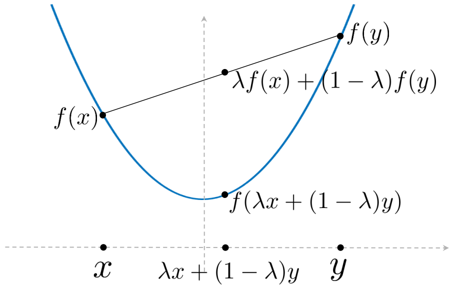


Figure 3: A geometric intuition behind a convex function.

What is the formal definition? The following observation can help us to easily come up with the definition. Take two points, say x and y, as in Fig. 3. Consider a point that lies in between the two points, say $\lambda x+(1-\lambda)y$ for $\lambda \in [0,1]$. Then, the bowl-shaped function suggests that the function evaluated at an λ-weighted linear combination of x and y is less than or equal to the same λ-weighted linear combination of the two functions evaluated at x and y:

$$f(\lambda x+(1-\lambda)y) \leq \lambda f(x)+(1-\lambda)f(y). \tag{21}$$

This motivates the following definition. We say that a function f is convex if (21) holds for all $\lambda \in [0,1]$ and for all x and y.

Appendix #2: Proof of convexity

Using the definition of convex optimization, one can prove that $J(w)$ is convex in optimization variable w. First we can readily show that convexity preserves under addition (why? think about the definition of convex functions). So it suffices to prove the following two:

$$\text{(i)} -\log \frac{1}{1+e^{-w^T x}} \text{ is convex in } w;$$

$$\text{(ii)} -\log \frac{e^{-w^T x}}{1+e^{-w^T x}} \text{ is convex in } w.$$

Since the second function in the above can be represented as the sum of a linear function and the first function:

$$-\log \frac{e^{-w^T x}}{1+e^{-w^T x}} = w^T x - \log \frac{1}{1+e^{-w^T x}},$$

it suffices to prove the convexity of the first function. Why?

Notice that the first function can be rewritten as:

$$-\log \frac{1}{1+e^{-w^T x}} = \log(1+e^{-w^T x}). \tag{22}$$

In fact, proving the convexity of (22) is a bit involved if one relies directly on the definition of convex functions. It turns out there is another way to prove. That is based on the computation of the second derivative of a function, called the Hessian. How to compute the Hessian? What is the dimension of the Hessian? For a function $f : \mathbf{R}^d \to \mathbf{R}$, the gradient $\nabla f(x) \in \mathbf{R}^d$ and the Hessian $\nabla^2 f(x) \in \mathbf{R}^{d\times d}$. If you are not familiar, check from the vector calculus course or from Wikipedia.

A well-known fact says that if the Hessian of a function is positive semi-definite (PSD)[1], then the function is convex. We will not prove this here. Don't worry about the proof, but do remember this fact. The statement itself is very instrumental. Here we will use this fact to prove the convexity of the function (22).

Taking a derivative of the RHS formula in (22) w.r.t. w, we get:

$$\nabla_w \log(1 + e^{-w^T x}) = \frac{-xe^{-w^T x}}{1 + e^{-w^T x}}.$$

This is due to a chain rule of derivatives and the fact that $\frac{d}{dz}\log z = \frac{1}{z}$, $\frac{d}{dz}e^z = e^z$ and $\frac{d}{dw}w^T x = x$. Taking another derivative of the above, we obtain a Hessian as follows:

$$\begin{aligned} \nabla_w^2 \log(1 + e^{-w^T x}) &= \nabla_w \left(\frac{-xe^{-w^T x}}{1 + e^{-w^T x}} \right) \\ &\overset{(a)}{=} \frac{xx^T e^{-w^T x}(1 + e^{-w^T x}) - xx^T e^{-w^T x} e^{-w^T x}}{(1 + e^{-w^T x})^2} \\ &= \frac{xx^T e^{-w^T x}}{(1 + e^{-w^T x})^2} \\ &\succeq 0 \end{aligned} \tag{23}$$

where (a) is due to the derivative rule of a quotient of two functions: $\frac{d}{dz}\frac{f(z)}{g(z)} = \frac{f'(z)g(z) - f(z)g'(z)}{g^2(z)}$. Here you may wonder why $\frac{d}{dw}(-xe^{-w^T x}) = xx^T e^{-w^T x}$. Why not xx, $x^T x^T$ or $x^T x$ in front of $e^{-w^T x}$? One rule-of-thumb that I strongly recommend is to simply try all the candidates and choose the one which does not have a syntax error (matrix dimension mismatch). For instance, xx (or $x^T x^T$) is just an invalid operation. $x^T x$ is not a right one because the Hessian must be an d-by-d matrix. The only candidate left without any syntax error is xx^T! We see that xx^T has the single eigenvalue of $\|x\|^2$ (Why?). Since the eigenvalue $\|x\|^2$ is non-negative, the Hessian is PSD, and therefore we prove the convexity.

[1] We say that a *symmetric* matrix, say $Q = Q^T \in \mathbf{R}^{d\times d}$, is positive semi-definite if $v^T Q v \geq 0$, $\forall v \in \mathbf{R}^d$, i.e., all the eigenvalues of Q are non-negative. It is simply denoted by $Q \succeq 0$.

EE210 Probability and Introductory Random Processes May 27, 2021
KAIST, Spring 2021 Changho Suh (chsuh@kaist.ac.kr)

Lecture 23: Speech recognition: Probabilistic modeling

Recap

During the past two lectures, we learned about a connection between machine learning and probability of this course's interest. The connection was made through the key fuel employed in a machine learning model: *data* $\{(x^{(i)}, y^{(i)})\}_{i=1}^m$. Since the data can be different depending on how we collect in a variety of contexts, it can be interpreted as a particular realization of a random process $\{(X^{(i)}, Y^{(i)})\}_{i=1}^m$, and this is where the connection to probability is made. We also demonstrated the role of the ML principle in the design of a perceptron-based optimization for machine learning:

$$\min_w \sum_{i=1}^m \ell(y^{(i)}, \hat{y}^{(i)}) \tag{1}$$

where $\ell(\cdot,\cdot)$ denotes a loss function and $\hat{y}^{(i)} := f_w(x^{(i)})$ indicates the prediction parameterized by the weights w. For activation, we considered the logistic function that implements a smooth transition of the step function:

$$f_w(x) = \frac{1}{1+e^{-w^T x}}. \tag{2}$$

We then showed that the optimal loss function in a sense of maximizing the likelihood function is *cross entropy loss*:

$$\ell^*_{\mathsf{CE}}(y, \hat{y}) = -y \log \hat{y} - (1-y)\log(1-\hat{y}),$$

which leads to a very well-known classifier: *logistic regression.* For the rest of this course, we will do the same thing yet for the last application: *speech recognition.*

Today's lecture

Today we will focus on the relationship between speech recognition and probability. Specifically what we are going to do are four-folded. First off, we will review the definition of speech recognition, and investigate its corresponding system. We will then figure out details on the input and the output of the system. Next, we will study the detailed structure of the system, and will develop a *probabilistic* model for the system. Based on the probabilistic model, we will finally show that speech recognition is an *inference* problem.

Review: Speech recognition

Let us recall what speech recognition is. A person speaks into a microphone, which samples the analog sound waveform. The goal of speech recognition is to figure out what speech (spoken words) means. In other words, it is to transform the analog waveform (comprising spoken words) into a written command, which can then be represented in the form of a *text.* So what we wish to decode in speech recognition is a text.

Speech recognition system

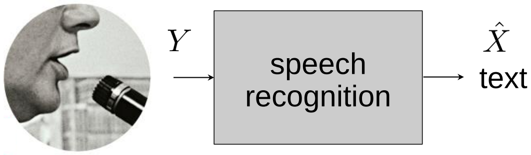


Figure 1: Speech recognition: Transforming voice signals into a written text.

Later we will show that speech recognition is an *inference* problem. So as an effort to introduce the interested entity X in front of the picture in Fig. 1, we consider a speech recognition *system*, as illustrated in Fig. 2. Notice that the input to the system is X that we wish to infer. Now

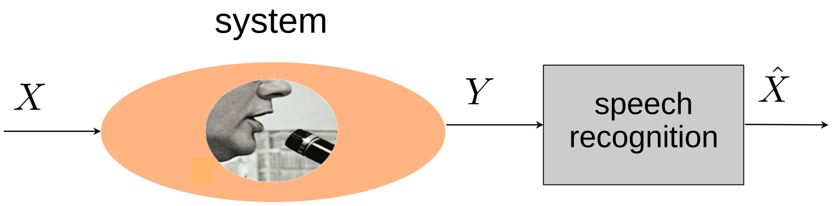
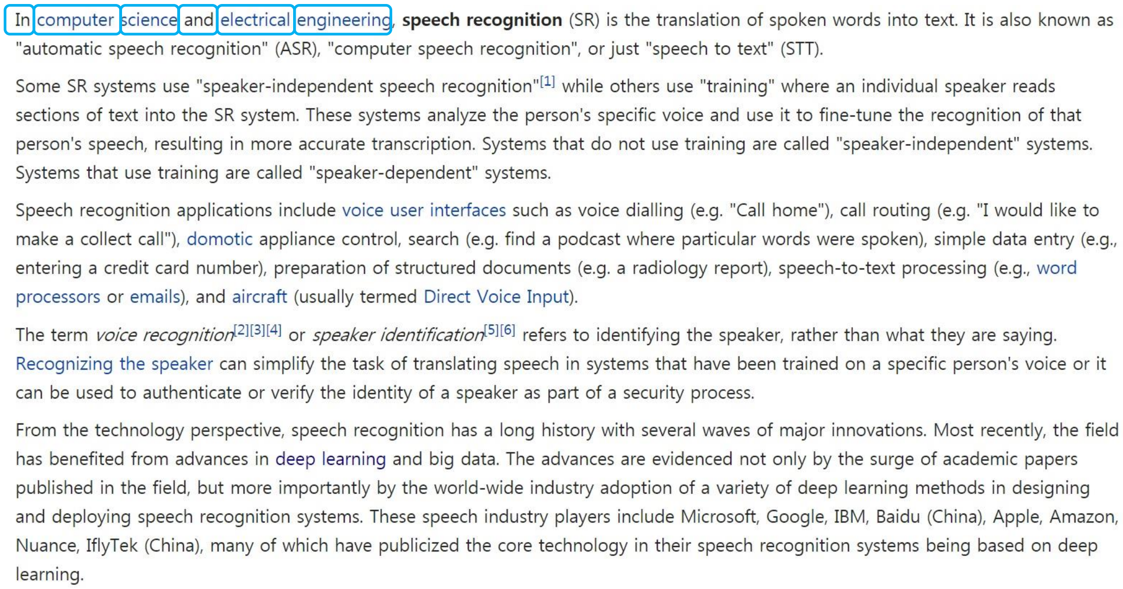


Figure 2: Speech recognition system.

what are detailed structures of the input X and the output Y in the system? To figure these out, let us first think about components that constitute a text. Obviously the components depend on a spoken language. Here we will consider the English language.

Structure of an English text



In computer science and electrical engineering, **speech recognition** (SR) is the translation of spoken words into text. It is also known as "automatic speech recognition" (ASR), "computer speech recognition", or just "speech to text" (STT).

Some SR systems use "speaker-independent speech recognition"[1] while others use "training" where an individual speaker reads sections of text into the SR system. These systems analyze the person's specific voice and use it to fine-tune the recognition of that person's speech, resulting in more accurate transcription. Systems that do not use training are called "speaker-independent" systems. Systems that use training are called "speaker-dependent" systems.

Speech recognition applications include voice user interfaces such as voice dialling (e.g. "Call home"), call routing (e.g. "I would like to make a collect call"), domotic appliance control, search (e.g. find a podcast where particular words were spoken), simple data entry (e.g., entering a credit card number), preparation of structured documents (e.g. a radiology report), speech-to-text processing (e.g., word processors or emails), and aircraft (usually termed Direct Voice Input).

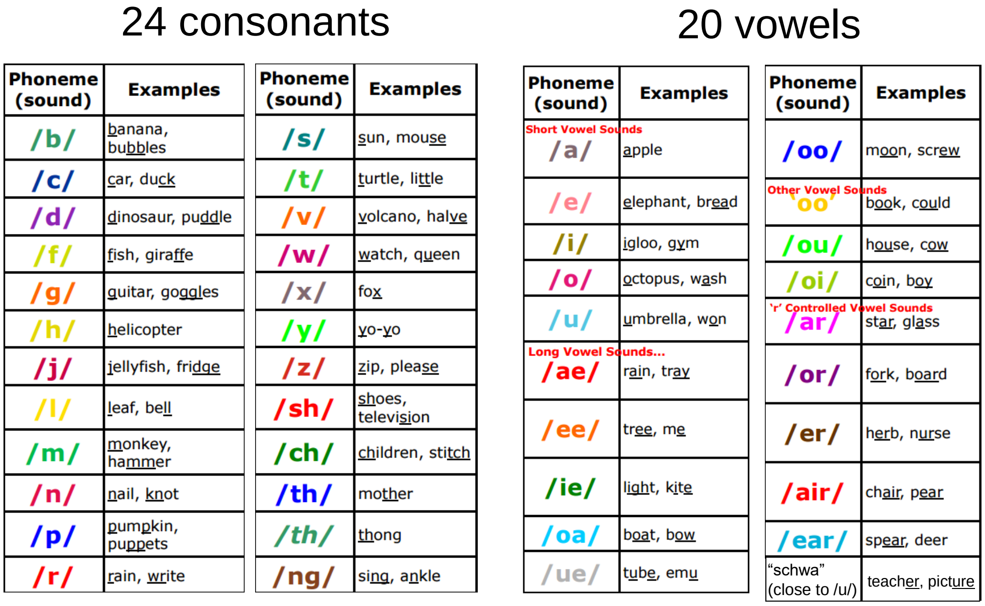
The term *voice recognition*[2][3][4] or *speaker identification*[5][6] refers to identifying the speaker, rather than what they are saying. Recognizing the speaker can simplify the task of translating speech in systems that have been trained on a specific person's voice or it can be used to authenticate or verify the identity of a speaker as part of a security process.

From the technology perspective, speech recognition has a long history with several waves of major innovations. Most recently, the field has benefited from advances in deep learning and big data. The advances are evidenced not only by the surge of academic papers published in the field, but more importantly by the world-wide industry adoption of a variety of deep learning methods in designing and deploying speech recognition systems. These speech industry players include Microsoft, Google, IBM, Baidu (China), Apple, Amazon, Nuance, IflyTek (China), many of which have publicized the core technology in their speech recognition systems being based on deep learning.

Figure 3: Structure of an English text.

A text is composed of a sequence of words, so one can view each word as a natural unit which we can decompose the text into. Actually we can further decompose the words into smaller units. For instance, consider a word "*speech*". One natural smaller unit that one can think of is an English alphabet (such as "s")? But there is an issue in adopting such a unit. The issue comes from the fact that an actual input to the speech recognition block is something related to *actual sound*, but the mapping between sound and alphabet is not one-to-one. For example, the sound /i/ is mapped to the letter "e" when we say "nike", but the same sound may refer to a different letter "i" when we say "bit". Here the slash indicates a conventional notation used for representing phonemes ("eumso" in Korean).

On the other hand, from a phonetic point of view, a word is decomposed into phonemes. For example, the word "*speech*" consists of four phonemes: /s/, /p/, /i/ and /ch/. The phoneme can well serve as the smallest phonetic unit in a language. There are two types of phonemes: (1) consonants; (2) vowels. There are 44 phonemes (24 consonants and 20 vowels) in English. See Fig. 4.



24 consonants

Phoneme (sound)	Examples
/b/	banana, bubbles
/c/	car, duck
/d/	dinosaur, puddle
/f/	fish, giraffe
/g/	guitar, goggles
/h/	helicopter
/j/	jellyfish, fridge
/l/	leaf, bell
/m/	monkey, hammer
/n/	nail, knot
/p/	pumpkin, puppets
/r/	rain, write

Phoneme (sound)	Examples
/s/	sun, mouse
/t/	turtle, little
/v/	volcano, halve
/w/	watch, queen
/x/	fox
/y/	yo-yo
/z/	zip, please
/sh/	shoes, television
/ch/	children, stitch
/th/	mother
/th/	thong
/ng/	sing, ankle

20 vowels

Phoneme (sound)	Examples
Short Vowel Sounds /a/	apple
/e/	elephant, bread
/i/	igloo, gym
/o/	octopus, wash
/u/	umbrella, won
Long Vowel Sounds... /ae/	rain, tray
/ee/	tree, me
/ie/	light, kite
/oa/	boat, bow
/ue/	tube, emu

Phoneme (sound)	Examples
/oo/	moon, screw
Other Vowel Sounds 'oo'	book, could
/ou/	house, cow
/oi/	coin, boy
'r' Controlled Vowel Sounds /ar/	star, glass
/or/	fork, board
/er/	herb, nurse
/air/	chair, pear
/ear/	spear, deer
"schwa" (close to /u/)	teacher, picture

Figure 4: 44 phonemes in English: (1) 24 consonants; (2) 20 vowels.

In light of this, the speech recognition problem can be viewed as the problem of figuring out the sequence of phonemes that forms a text. See Fig. 5.

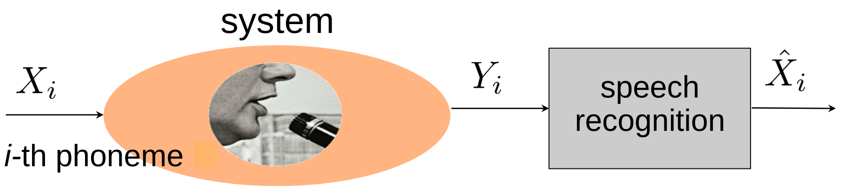


Figure 5: Speech recognition system: Each component of the input X_i indicates the ith phoneme that takes one of 44 phonemes (24 consonants and 20 vowels).

The sequence $\{X_i\}_{i=1}^n$ of phonemes is the one that we wish to decode, so the phonemes can be considered as random variables. Here we let $X_i \in \mathcal{X}$ be the ith phoneme of $\{X_i\}_{i=1}^n$ where $\mathcal{X}$ is the set of phonemes whose alphabet size is $|\mathcal{X}| = 44$.

Inside the system

Now let us figure out what the output Y_i is in the speech recognition system? The output $\{Y_i\}_{i=1}^n$ is the one that will be put into the speech recognition block, so it should reflect something related to actual sound. To figure out what it is, we first need to relate $\{X_i\}_{i=1}^n$ to the actual sound signal that will be picked up at the microphone in the system. See Fig. 6. A user who desires to say what the text $\{X_i\}_{i=1}^n$ means speaks corresponding spoken words into the microphone, generating the analog waveform, say $Y(t)$. Actually, instead of the analog continuous-time signals $Y(t)$, we want *discrete*-time quantities that can be represented as a sequence $\{Y_i\}_{i=1}^n$. It turns out there is a way to translate $Y(t)$ into discrete-time signals.

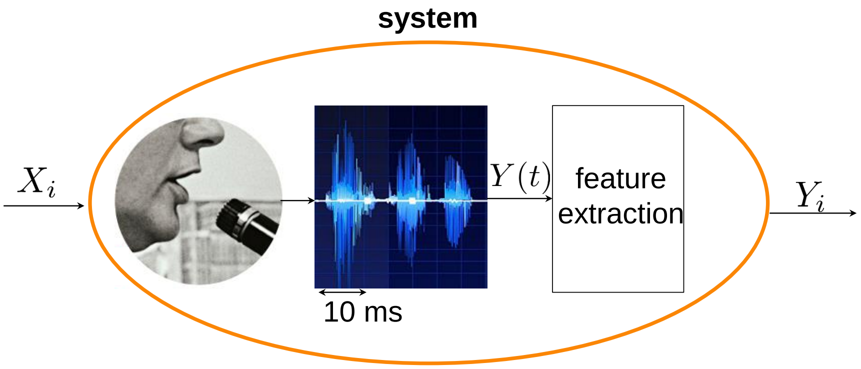


Figure 6: Inside the speech recognition system.

The translation is based on the following observation. Each phoneme spans roughly 10 ms, as in Fig. 6. Of course, it can vary depending on the speaking pace of an individual. The 10 ms is sort of an average taken across many phonemes from different people. We chop the analog waveform into 10 ms intervals. We then take the signal in each 10 ms interval and wish to extract some key components (discrete-time quantities) from it. Here for simplicity of discussion, we assume no pause between phonemes. In the presence of pauses, one can readily detect them and then chop the corresponding signals. It turns out that the relevant information contained in speech is most apparent in the *frequency* domain. So one natural approach that we can take is to employ a well-known transformation technique that highlights frequency components. That is, *Fourier transform*:

$$Y(f) := \int_{-\infty}^{+\infty} Y(t) e^{-j2\pi f t} dt. \tag{3}$$

However, we have some issues here. Two issues. One is that the input to Fourier transform is a *infinite-time-horizon* signal while we wish to extract a certain component that corresponds only to the signal w.r.t. each 10 ms time interval. One way to address this is to extract only a part of $Y(f)$ by taking a time-windowed Fourier transform:

$$Y_i(f) := \int_{10 \text{ ms}\cdot(i-1)}^{10 \text{ ms}\cdot i} Y(t) e^{-j2\pi f t} dt. \tag{4}$$

The second issue is that $Y_i(f)$ is still a *continuous* quantity in light of f. So one walk-around is to extract corresponding Fourier *coefficients* out of it, by taking $Y_i(f)$ at particular f's:

$$Y_i := \begin{bmatrix} Y_i(f_1) \\ Y_i(f_2) \\ \vdots \\ Y_i(f_k) \end{bmatrix} \tag{5}$$

where f_j's are certain frequencies that yield significant spectral components, and k is the number of such frequencies. As shown above, typically there are multiple Fourier coefficients for the signal in each 10 ms interval. But let us simplify the story by assuming there is only one significant spectral component (i.e., $k = 1$). Here we call that component a feature. We let Y_i be the ith feature with respect to the ith phoneme. See Fig. 6 for the entire procedure inside the system.

Relation between $\{X_i\}_{i=1}^n$ and $\{Y_i\}_{i=1}^n$

Fig. 7 illustrates the whole picture with X_i (the ith phoneme) and Y_i (the ith feature). Now how do $\{X_i\}_{i=1}^n$ and $\{Y_i\}_{i=1}^n$ relate with each other? There is a lot of randomness involved in the system. Two major sources of the randomness are: (1) different voice characteristics (e.g., accent) and (2) noise (e.g., thermal noise due to random movements of electrons in the electrical circuit). This randomness induces uncertainty in $\{Y_i\}_{i=1}^n$, making the input and the output related probabilistically. Hence, we can view speech recognition as an *inference* problem.

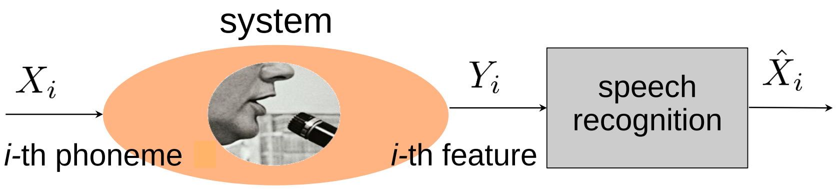


Figure 7: Speech recognition system: X_i indicates the ith phoneme and Y_i denotes the corresponding feature (spectral information).

Look ahead

Since speech recognition is an inference problem, the optimal inference is again *MAP*. So next time, we will investigate the optimal MAP estimator for speech recognition.

Lecture 24: Speech recognition: MAP principle

Recap

Last time we showed that speech recognition is an inference problem wherein the goal is to infer a sequence $\{X_i\}_{i=1}^n$ of phonemes (that allows us to recognize what a speaker says) from a sequence $\{Y_i\}_{i=1}^n$ of features (that have bearing on actual spoken words). See Fig. 1 for details.

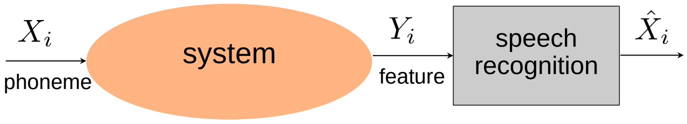


Figure 1: A block diagram of the speech recognition system and recovery block

We also found that there is a lot of randomness in the system due to different voice characteristics of a certain speaker and the system noise (e.g, thermal noise). The randomness is the one that makes the output of the system (observation) probabilistically related to the input (an interested entity for inference), demonstrating that speech recognition problem is indeed an inference problem.

Today's lecture

Today we will investigate the optimal algorithm based on MAP. It consists of three parts. First we will derive the optimal MAP estimator. As we know, in order to obtain an explicit MAP solution, we need to succinctly represent the "a priori probability" and the "likelihood". It turns out there is a nice statistical structure on $\{(X_i, Y_i)\}_{i=1}^n$ that enables an efficient representation of the two quantities. So in the second part, we will explore the nice structure. Lastly we will exploit the structure to simplify the MAP estimator.

Optimal algorithm for speech recognition

Since speech recognition is an inference problem, the optimal inference is the one that maximizes conditional correct-decision probability:

$$\mathbb{P}(X_1 = \hat{X}_1, \ldots, X_n = \hat{X}_n | Y_1 = y_1, \ldots, Y_n = y_n). \tag{1}$$

As we figured out several times, it coincides with the MAP rule which finds the one that maximizes the a posteriori probability:

$$\begin{aligned}
\hat{\mathbf{x}}_{\mathsf{MAP}} &= \arg \max_{\hat{x}_1, \ldots, \hat{x}_n \in \mathcal{X}} \mathbb{P}(X_1 = \hat{x}_1, \ldots, X_n = \hat{x}_n | Y_1 = y_1, \ldots, Y_n = y_n) \\
&= \arg \max_{\hat{x}_1, \ldots, \hat{x}_n \in \mathcal{X}} \frac{\mathbb{P}_{\mathbf{X}}(\hat{x}_1, \ldots, \hat{x}_n) f(y_1, \ldots, y_n | \hat{x}_1, \ldots, \hat{x}_n)}{f(y_1, \ldots, y_n)} \\
&= \arg \max_{\hat{x}_1, \ldots, \hat{x}_n \in \mathcal{X}} \mathbb{P}_{\mathbf{X}}(\hat{x}_1, \ldots, \hat{x}_n) f(y_1, \ldots, y_n | \hat{x}_1, \ldots, \hat{x}_n)
\end{aligned} \tag{2}$$

when the 2nd equality follows from the definition of conditional probability and the fact that

y_i's are continuous values. Here $\mathcal{X}$ denotes a set of all the phonemes that each X_i can take on, usually called the alphabet.

Two quantities that we need to know about

To compute the MAP solution (2), we need to figure out two quantities: (1) A priori probability: $\mathbb{P}_{\mathbf{X}}(\hat{x}_1, \ldots, \hat{x}_n)$; (2) conditional pdf (likelihood): $f(y_1, \ldots, y_n | \hat{x}_1, \ldots, \hat{x}_n)$, which captures the statistical relationship between the input and the output. Just a side note: In light of a communication system, the speech recognition system can be interpreted as a *channel*.

A naive approach to obtain $\mathbb{P}_{\mathbf{X}}(\hat{x}_1, \ldots, \hat{x}_n)$ is investigating the quantity for each sequence. However, this simple way comes with a challenge in complexity. The reason is that the number of possible patterns of the sequence $\{X_i\}_{i=1}^n$ grows exponentially with n:

$$|\mathcal{X}|^n = 44^n. \tag{3}$$

Remember that there are 44 phonemes in English: 24 consonants and 20 vowels. The total number of probability values required to fully specify $\mathbb{P}_{\mathbf{X}}(\hat{x}_1, \ldots, \hat{x}_n)$ is huge especially for a large n, rendering it challenging to obtain the a priori knowledge. Also there are many different values for the likelihood $f(y_1, \ldots, y_n | \hat{x}_1, \ldots, \hat{x}_n)$. Even worse, $(y_1, \ldots, y_n)$ are continuous values.

But it turns out there is a very nice statistical structure on $\{(X_i, Y_i)\}_{i=1}^n$ that enables an efficient representation of the a priori probability and the likelihood. Exploiting the statistical structure, we are able to model $\{(X_i, Y_i)\}_{i=1}^n$ for which $\mathbb{P}_{\mathbf{X}}(\hat{x}_1, \ldots, \hat{x}_n)$ and $f(y_1, \ldots, y_n | \hat{x}_1, \ldots, \hat{x}_n)$ can be specified with a much smaller number of parameters.

A random process $\{X_i\}_{i=1}^n$

One naive way of modeling the sequence of phonemes is assuming that these random variables are independent. We saw an independent process before many times. One example was the additive white Gaussian noise that we discussed in the communication application. It is an i.i.d. random process. However, the independence assumption is not relevant in speech recognition. Some phonemes are more likely to follow other phonemes. For instance, the phoneme /th/ is more likely to be followed by /e/ rather than /s/. So assuming the random variables to be independent seems like a pretty bad idea.

A generalized Markov model

How are $\{X_i\}_{i=1}^n$ related then? It turns out a random process that we learned about in Lecture 12 can well capture the dependency across phonemes. That is, the *generalized Markov model.* Remember its definition. We say that $\{X_i\}_{i=1}^n$ is a generalized Markov process with ℓ memories if

$$\mathbb{P}(x_{i+1} | x_i, \ldots, x_{i-\ell+1}, x_{i-\ell}, \ldots, x_1) = \mathbb{P}(x_{i+1} | x_i, \ldots, x_{i-\ell+1}).$$

We also checked that another properly defined random process $S_i := (X_i, \ldots, X_{i-\ell+1})$ is a single-memory Markov process. So it suffices to focus on the case $\ell = 1$ with a proper rearrangement. Just for illustrative purpose, we will assume that the sequence of phonemes is a single-memory Markov process.

Joint distribution $\mathbb{P}_{\mathbf{X}}(x_1, \ldots, x_n)$

Using the graphical model that we learned in Lecture 12, we can represent the single-memory Markov process as:

$$X_1 - X_2 - X_3 - \cdots - X_{n-1} - X_n. \tag{4}$$

Recall it is called a Markov chain as it looks like a chain. Using this statistical structure, one can now write down the joint distribution as:

$$\begin{aligned}
&\mathbb{P}_{\mathbf{X}}(x_1,\ldots,x_n)\\
&\overset{(a)}{=} \mathbb{P}(x_1)\mathbb{P}(x_2|x_1)\mathbb{P}(x_3|x_2,x_1)\cdots\mathbb{P}(x_n|x_{n-1},\ldots,x_1)\\
&\overset{(b)}{=} \mathbb{P}(x_1)\mathbb{P}(x_2|x_1)\mathbb{P}(x_3|x_2)\cdots\mathbb{P}(x_n|x_{n-1})\\
&= \mathbb{P}(x_1)\prod_{i=2}^{n}\mathbb{P}(x_i|x_{i-1})
\end{aligned} \tag{5}$$

where (a) is due to the definition of conditional probability; and (b) comes from the Markov property. Hence, it suffices to know about only $\mathbb{P}(x_1)$ and $\mathbb{P}(x_i|x_{i-1})$ to compute $\mathbb{P}_{\mathbf{X}}(x_1,\ldots,x_n)$.

How to figure out $\mathbb{P}(x_1)$ and $\mathbb{P}(x_i|x_{i-1})$?

We only need to specify 44 values for $\mathbb{P}(x_1)$ and 44^2 values for $\mathbb{P}(x_i|x_{i-1})$. The sum $44+44^2$ is much smaller than the huge number 44^n required to specify the joint distribution when the statistical structure is not exploited. In reality, one can estimate individual pmf $\mathbb{P}(x_1)$ and the transition probability $\mathbb{P}(x_i|x_{i-1})$ from any large text by computing the following sample means:

$$\mathbb{P}(/\mathrm{s}/) = \mathbb{P}(X_1 = /\mathrm{s}/) \approx \frac{\text{\# of occurences of ``}s\text{''}}{\text{\# of phonemes in the interested text}}; \tag{6}$$

$$\mathbb{P}(/\mathrm{t}/|/\mathrm{s}/) = \mathbb{P}(X_i = /\mathrm{t}/|X_{i-1} = /\mathrm{s}/) \approx \frac{\text{\# of ``}t\text{'' that follows ``}s\text{''}}{\text{\# of occurrences of ``}s\text{''}}. \tag{7}$$

It turns out the law of large numbers (LLN) that we learned about w.r.t. the i.i.d. random process can be extended to the Markov process as well. We will not prove it, as the proof distracts the current storyline (also the proof is not that simple). If you want to know more, you may want to take a course on random processes. By using the extended LLN, the estimates in the above become concentrated around the ground-truth distributions as the number of phonemes in the text grows.

Likelihood function $f(y_1,\ldots,y_n|x_1,\ldots,x_n)$

Now let us figure out how to obtain the knowledge on the likelihood. We find that a key observation on the system enables us to identify the statistical structure of $\{Y_i\}_{i=1}^n$, thus providing a concrete way of computing $f(y_1,\ldots,y_n|x_1,\ldots,x_n)$. Recall the inside of the system; see Fig. 2. Here the key observation is that Y_i can be viewed as a noisy version of X_i and the noise has nothing to do with any other random variables involved in the system. The mathematical representation of the observation is that given X_i, Y_i is mutually independent of all the other random

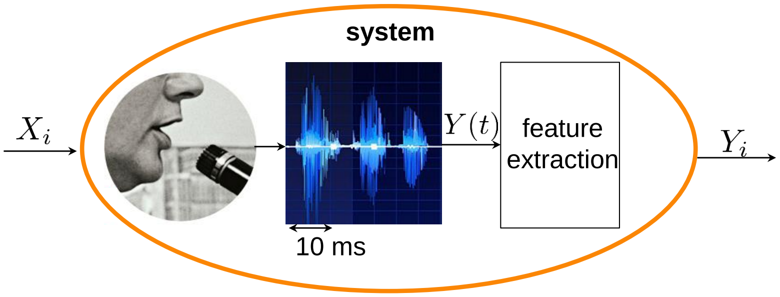


Figure 2: Inside of the speech recognition system.

variables: e.g.,

$$Y_1 \perp (X_2, \ldots, X_n, Y_2, \ldots, Y_n) | X_1$$

where the symbol $\perp$ means "mutual independence". This property leads to the graphical model for $\{Y_i\}_{i=1}^n$ as in Fig. 3.

$$\begin{array}{ccccccccccc} Y_1 & & Y_2 & & Y_2 & & & & Y_{n-1} & & Y_n \\ | & & | & & | & & & & | & & | \\ X_1 & - & X_2 & - & X_3 & - & \cdots & - & X_{n-1} & - & X_n \end{array}$$

Figure 3: A Hidden Markov Model (HMM) for the output $\{Y_i\}_{i=1}^n$ of the speech recognition system.

Notice that if we remove the node X_i, the node Y_i will be disconnected from the rest of the graph. This reflects the fact that Y_i depends on other random variables only through X_i. One interesting question about the model: Is the observation sequence $\{Y_i\}_{i=1}^n$ a Markov model? No! Why? But the *underlying sequence* $\{X_i\}_{i=1}^n$ that we want to figure out is a Markov model. That's why it is called the *Hidden Markov Model*, HMM for short.

Using this statistical property, one can write down the conditional pdf as:

$$\begin{aligned} & f(y_1, \ldots, y_n | x_1, \ldots, x_n) \\ & = f(y_1 | x_1, \ldots, x_n) f(y_2, \ldots, y_n | x_1, \ldots, x_n, y_1) \\ & \overset{(a)}{=} f(y_1 | x_1) f(y_2, \ldots, y_n | x_1, \ldots, x_n, y_1) \\ & \qquad \vdots \\ & \overset{(b)}{=} f(y_1|x_1) f(y_2|x_2) \cdots f(y_n|x_n) \\ & = \prod_{i=1}^{n} f(y_i|x_i) \end{aligned} \tag{8}$$

where (a) and (b) are because given x_i, y_i is independent of everything else for any i.

Note that it suffices to know only $f(y_i|x_i)$ to compute $f(y_1, \ldots, y_n | x_1, \ldots, x_n)$. Now a question is: How can we obtain the knowledge of the individual likelihood function $f(y_i|x_i)$. If the ith feature y_i is a *discrete* value, then we need to specify only the number $|\mathcal{X}| \times |\mathcal{Y}|$ of possible values for $f(y_i|x_i)$. But the value of the feature is in general a *continuous* value although it can be quantized so that it can be represented by a discrete random variable. So we need to figure out the *functional relationship* between y_i and x_i to specify the likelihood function.

How to figure out $f(y_i|x_i)$?

Remember that the likelihood function depends on *randomness* that occurs in the system. The randomness comes from the following two: (i) different voice characteristics; and (ii) a noise introduced in an electrical circuit employed in the system. If the system has no noise and a speaker is given, we can think of Y_i simply as a deterministic function of X_i. For example, given $X_i = /a/$, Y_i is a deterministic function of $/a/$, say μ_a. See Fig. 4. However, due to the noise in the system, the feature Y_i would be randomly distributed around the true feature μ_a corresponding to the phoneme $/a/$. Remember what we learned about the noise in the communication application. The major source of the noise is the random movement of electrons

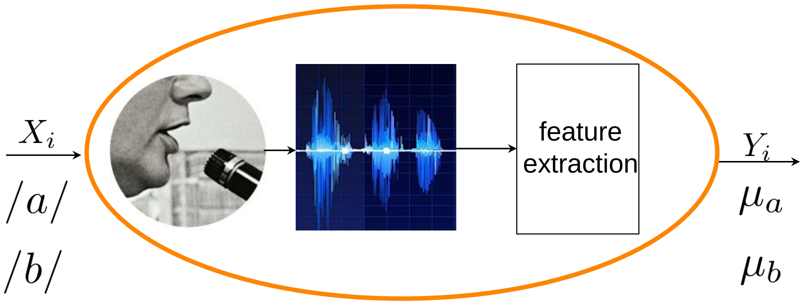


Figure 4: Relationship between X_i and Y_i when a speaker is given in the noiseless case.

due to heat. So it is the thermal noise. We know that the thermal noise can be modeled as an additive white Gaussian noise (AWGN). Hence, given $X_i = /a/$, we can model Y_i as:

$$Y_i = \mu_a + Z_i \tag{9}$$

where Z_i's are i.i.d. $\sim \mathcal{N}(0, \sigma^2)$. Similarly given $X_i = /b/$, Y_i can be modeled as the true feature concerning $/b/$, say μ_b, plus an additive Gaussian noise. Here for simplicity, we will assume that we know the variance of the additive noise σ^2. One way to estimate the variance is via MLE for Gaussian distribution that we learned in Lecture 16. That is, to infer the variance from multiple measurements Y_i's fed by null signals. More concretely, suppose $X_i = 0$ for $i = 1, \ldots, n$. Then, under the statistical modeling on Y_i as above, we get:

$$Y_i = Z_i, \quad i = 1, \ldots, n. \tag{10}$$

Then the sample mean of Y_i^2's is the MLE for σ^2:

$$\sigma^2_{\sf ML} = \frac{Y_1^2 + \cdots + Y_n^2}{n}. \tag{11}$$

Even if we know the noise variance, the likelihood function $f(y_i|x_i)$ is not specified yet. This is because the true features such as μ_a and μ_b are *user-specific* parameters. Hence, these need to be estimated to fully specify the likelihood function. There is one very popular approach that enables the estimation. We will next study the approach.

Machine learning for estimating (μ_a, μ_b)

Assume for a moment that we knew the sequence of phonemes. This sounds like an absurd assumption given that the sequence of phonemes is exactly what we would like to infer. However, we could ask the speaker to say some predetermined phonemes for us at the beginning. In fact, some speech recognition software does this. Then we could use this sequence to estimate μ_a and μ_b based on the known phonemes. Note in this method that we are learning the parameters by asking the speaker to provide input-output example pairs at the beginning. What does this remind you of? Yes, it is *machine learning*.

Here is how machine learning works in detail. Suppose we ask the user to say "*aaaaaaaa*" for eight time slots. This then gives us:

$$Y_i = \mu_a + Z_i, \quad i = 1, \ldots, 8. \tag{12}$$

Since the user is given, μ_a is fixed, and therefore $Y_i \sim \mathcal{N}(\mu_a, \sigma^2)$. Here the goal is to estimate μ_a given eight i.i.d. samples $(Y_1, \ldots, Y_8)$. Again what does this remind you of? Yes, the optimal way to estimate μ_a is the MLE:

$$\hat{\mu}_a^{\mathsf{ML}} = \frac{Y_1 + \cdots + Y_8}{8}. \tag{13}$$

Simplified optimal algorithm

Applying (5) and (8) into the objective function in (2), we get:

$$\mathbb{P}_{\mathbf{X}}(x_1, \ldots, x_n) f(y_1, \ldots, y_n | x_1, \ldots, x_n) = \mathbb{P}(x_1) \prod_{i=2}^{n} \mathbb{P}(x_i | x_{i-1}) \prod_{i=1}^{n} f(y_i | x_i).$$

Plugging this into (2), we obtain:

$$\hat{\mathbf{x}}_{\mathsf{MAP}} = \arg \max_{\hat{x}_1, \ldots, \hat{x}_n \in \mathcal{X}} \mathbb{P}(x_1) \prod_{i=2}^{n} \mathbb{P}(x_i | x_{i-1}) \prod_{i=1}^{n} f(y_i | x_i). \tag{14}$$

Look ahead

It turns out there is an efficient way to compute the simplified MAP solution (14). Next time, we will explore the way.

EE210 Probability and Introductory Random Processes June 3, 2021
KAIST, Spring 2021 Changho Suh (chsuh@kaist.ac.kr)

Lecture 25: Speech recognition: Viterbi algorithm I

Recap

During the past two lectures, we showed that speech recognition is an inference problem and then derived the optimal MAP estimator. We also simplified the MAP estimator by exploring the nice statistical structure on $\{(X_i, Y_i)\}_{i=1}^n$. Here X_i indicates the ith phoneme and Y_i denotes the ith feature (spectral information). See Fig. 1. The MAP estimator reads:

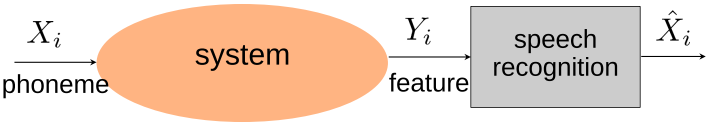


Figure 1: A block diagram of the speech recognition system and recovery block

$$\hat{\mathbf{x}}_{\mathsf{MAP}} = \arg \max_{x_1,\ldots,x_n \in \mathcal{X}} \mathbb{P}_{\mathbf{X}}(x_1,\ldots,x_n) f(y_1,\ldots,y_n|x_1,\ldots,x_n). \quad (1)$$

Assuming a single-memory Markov model for $\{X_i\}_{i=1}^n$, we simplified the a priori probability as:

$$\mathbb{P}_{\mathbf{X}}(x_1,\ldots,x_n) = \mathbb{P}(x_1)\prod_{i=2}^n \mathbb{P}(x_i|x_{i-1}).$$

Exploiting the HMM structure of $\{Y_i\}_{i=1}^n$, we could figure out that the likelihood is the product of every individual:

$$f(y_1,\ldots,y_n|x_1,\ldots,x_n) = \prod_{i=1}^n f(y_i|x_i).$$

Plugging the above two into (1), we then obtained the simplified MAP estimator:

$$\hat{\mathbf{x}}_{\mathsf{MAP}} = \arg \max_{x_1,\ldots,x_n \in \mathcal{X}} \mathbb{P}(x_1)\prod_{i=2}^n \mathbb{P}(x_i|x_{i-1})\prod_{i=1}^n f(y_i|x_i). \quad (2)$$

One naive way to derive $\hat{\mathbf{x}}_{\mathsf{MAP}}$ is to search for all the possible sequence patterns that $\{x_i\}_{i=1}^n$ can take on. However, this way, called the *exhaustive search*, comes with a challenge. The challenge is that the total number of possible sequence patterns is huge, $|\mathcal{X}|^n = 44^n$. The complexity grows *exponentially* with n.

Today's lecture

For the remaining two lectures including today's one, we will study another very popular method that addresses the computational challenge. The method is called the *Viterbi algorithm*. Today we will investigate several concepts that form the basis of algorithm. Next time, we will figure out how the algorithm works in detail. This lecture consists of three parts. Actually the "Viterbi" is the last name of the inventor of the algorithm. First off, I will explain who the inventor is, and will emphasize the key feature of the algorithm w.r.t. complexity prior to diving into details. It turns out the Viterbi algorithm is a very *generic* algorithm with a wide variety of applications, yet being subject to a particular form of optimization. So in the second part, we will translate the original optimization (2) into another respecting the particular form. Lastly we will study three important concepts which serve to explain how the algorithm works.

The key feature of Viterbi algorithm

The inventor of the algorithm is Andrew Viterbi. See Fig. 2. He is one of the giants in the fields of communication and information theory, also known as a co-founder of Qualcomm Inc that you may hear of. Search for wikipedia. You will soon figure out he is indeed a giant figure. Actually he came up with the algorithm in the process of addressing a different yet interesting problem that arises in communication. Later, many people figured out the algorithm can be applicable to a widening array of problems beyond communication, including speech recognition of our current interest.

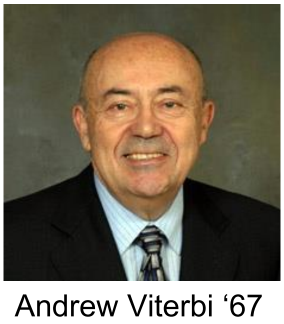


Figure 2: Andrew Viterbi is a giant in the communication and information theory fields.

The key feature of the algorithm that I want to emphasize is that the algorithm complexity grows *linearly* with n, being much smaller than 44^n that we faced with the exhaustive search. As you may figure out later, the algorithm has nothing to do with probability of this course's focus. As emphasized in the key feature, however, the algorithm is so powerful with extremely lower complexity. This is the sole reason that I want to introduce this algorithm to you guys. Please be familiar with the Viterbi algorithm in your entire career. The algorithm will constantly benefit you, as long as you are working on something relevant to optimization.

Translation into the canonical form

Now let's start by translating the original optimization (2) into another with the canonical structure that the algorithm relies upon. To this end, we first massage the optimization (2):

$$\hat{\mathsf{x}}_{\mathsf{MAP}} = \arg \max_{x_1,\ldots,x_n \in \mathcal{X}} \mathbb{P}(x_1) \prod_{i=2}^{n} \mathbb{P}(x_i|x_{i-1}) \prod_{i=1}^{n} f(y_i|x_i).$$

Setting $x_0 = 0$ (nothing), we can simplify the optimization as:

$$\begin{aligned}
\hat{\mathbf{x}}_{\mathsf{MAP}} &= \arg \max_{x_1,\ldots,x_n \in \mathcal{X}} \prod_{i=1}^{n} \mathbb{P}(x_i|x_{i-1}) f(y_i|x_i) \\
&= \arg \max_{x_1,\ldots,x_n \in \mathcal{X}} \sum_{i=1}^{n} \log \{\mathbb{P}(x_i|x_{i-1}) f(y_i|x_i)\} \\
&= \arg \min_{x_1,\ldots,x_n \in \mathcal{X}} \sum_{i=1}^{n} \underbrace{\log \left\{ \frac{1}{\mathbb{P}(x_i|x_{i-1}) f(y_i|x_i)} \right\}}_{=:c_i(s_i)}
\end{aligned}$$

where the 2nd equality comes from the fact that taking an increasing function $\log(\cdot)$ does not alter the maximizer; and the last equality is due to changing the sign of the objective function. Notice that the ith component in the summation in the last line is a function of (x_{i-1}, x_i, y_i). So one can denote the ith component by $c_i(s_i)$ where

$$s_i := \begin{bmatrix} x_{i-1} \\ x_i \end{bmatrix}. \tag{3}$$

Here we call s_i the *state*. Later you will figure out the rationale behind the naming. The reason we leave the subscript i in c is that the ith component is also a function of y_i. Notice that decoding $(x_1, x_2, \ldots, x_n)$ is equivalent to decoding $(s_1, s_2, \ldots, s_n)$. Also remember that y_i's are given and x_i's are optimization variables. So the original optimization (2) is equivalent to:

$$\hat{\mathbf{s}}_{\mathsf{MAP}} = \arg \min_{s_1,\ldots,s_n} \sum_{i=1}^{n} c_i(s_i) \tag{4}$$

where

$$c_i(s_i) = \log \left\{ \frac{1}{\mathbb{P}(x_i|x_{i-1}) f(y_i|x_i)} \right\}. \tag{5}$$

The above optimization (4) is the canonical optimization that the Viterbi algorithm is built upon. We are now ready to study the three concepts that form the basis of the algorithm.

Concept #1: Cost

The first concept is concerned about the interested quantity $c_i(s_i)$ in the canonical optimization (4). Here the quantity $c_i(s_i)$ can be viewed as something *negative*, because the smaller quantity, the better the situation is. Hence, we call it "cost". Now you can see why I use the notation c to denote the quantity.

Concept #2: State s_i

The second concept is the state s_i that we defined in (3). The state has certain properties. For ease of explanation of the properties, let us assume that x_i takes one among only two possible phonemes, say $/a/$ and $/b/$. For notational simplicity, let us drop the slash symbol: $x_i \in \{a, b\}$. In this case, we can readily see that s_i can take one of the following six candidates:

$$\begin{bmatrix} 0 \\ b \end{bmatrix}, \begin{bmatrix} 0 \\ a \end{bmatrix}, \underbrace{\begin{bmatrix} a \\ a \end{bmatrix}}_{s1}, \underbrace{\begin{bmatrix} b \\ a \end{bmatrix}}_{s2}, \underbrace{\begin{bmatrix} b \\ b \end{bmatrix}}_{s3}, \underbrace{\begin{bmatrix} a \\ b \end{bmatrix}}_{s4}. \tag{6}$$

Observe that the first two occur only at the beginning ($i = 1$), assuming that $x_0 = 0$. So the two states are negligible relative to the others for a large value of n. For simplification, let us not worry about the two states. To this end, we intentionally set $x_0 = a$. In this case, we have only four possible states. The number "four" is obviously finite. There is a terminology which indicates an entity w.r.t. such state that can take one among only the *finite* number of possible candidates. That is, a *Finite State Machine*, FSM for short.

The FSM has an interesting property which can be illustrated in picture. To see this, first observe that each state s_i can move from one to another, depending on the value of x_{i+1}. So one can now think of the following state translations as illustrated in Fig. 3. Here a or b labeled above a transition arrow indicates the value of x_{i+1}, given the current state s_i. For instance, suppose that $x_{i+1} = b$ given the current state s1. Then, we move along the red transition arrow to arrive at the state s4 of $[a; b]$. The picture like Fig. 3 is called the *state transition diagram*.

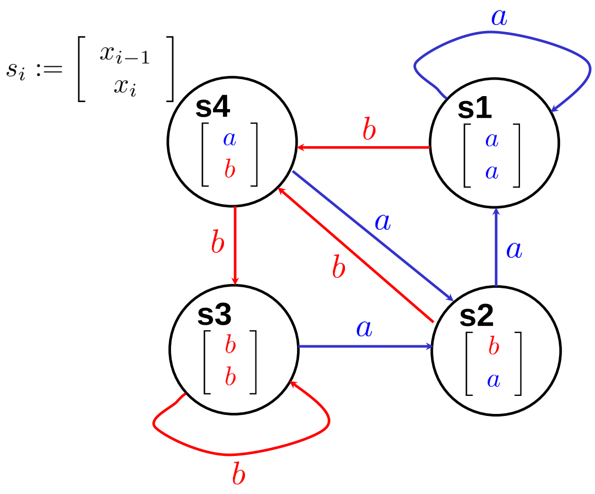

$s_i := \begin{bmatrix} x_{i-1} \\ x_i \end{bmatrix}$
s4
$\begin{bmatrix} a \\ b \end{bmatrix}$
s1
$\begin{bmatrix} a \\ a \end{bmatrix}$
s3
$\begin{bmatrix} b \\ b \end{bmatrix}$
s2
$\begin{bmatrix} b \\ a \end{bmatrix}$
a
b
a
a
b
b
a
b

Figure 3: A finite state machine with four states. A transition occurs depending on the value of x_{i+1}.

While the state transition diagram well represents how each state moves around to another, it does not capture the *time evolution*. This is exactly where another concept w.r.t. the FSM arises. That is, the *trellis diagram*. Below we will describe how the trellis diagram looks like and then will explain how it helps solve the interested optimization (4) that concerns the given information $\{y_i\}_{i=1}^n$.

Concept #3: Trellis diagram

The trellis diagram exhibits both state transitions and time evolution. To clearly understand how it works, let me walk you through details with the help of Figs. 4 and 5.

Consider the state at time 1:

$$s_1 = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} a \\ x_1 \end{bmatrix} = \begin{cases} \begin{bmatrix} a \\ a \end{bmatrix} = \text{s1}, & \text{if } x_1 = a; \\ \begin{bmatrix} a \\ b \end{bmatrix} = \text{s4}, & \text{if } x_1 = b, \end{cases}$$

where the second equality is due to our assumption $x_0 = a$. Notice that s_1 takes one of the two possible states s1 and s4, reflected in the two dots at time 1 in Fig. 4. In time 0, we have two options for s_0 depending on the value of x_{-1}: s1 and s2. To remove the ambiguity in time 0, let

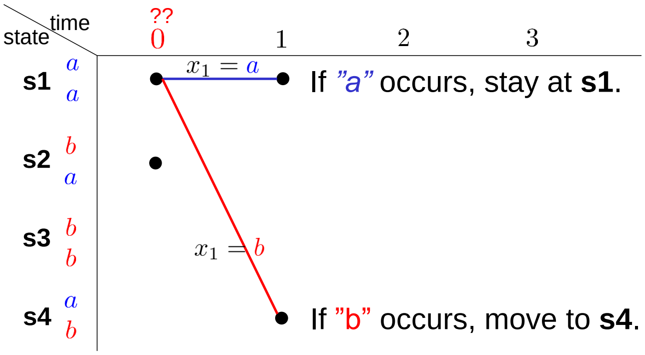


Figure 4: A trellis diagram at $i = 1$.

us further assume $x_{-1} = a$. This way, we can fix the initial state as s1. So we ensure that we start always from s1. As mentioned earlier, if $x_1 = a$, then $s_1 = \text{s1}$; otherwise $s_1 = \text{s4}$.

To be more familiar with how it works, let us consider one more time slot as illustrated in Fig. 5. Suppose $s_1 = \text{s1}$. If $x_2 = a$, then it stays at the same state s1 (reflected in the blue transition arrow); otherwise, it moves to s4 (reflected in the red arrow). On the other hand, given $s_1 = \text{s4}$, if $x_2 = a$, then it moves to s2; otherwise, it goes to s3.

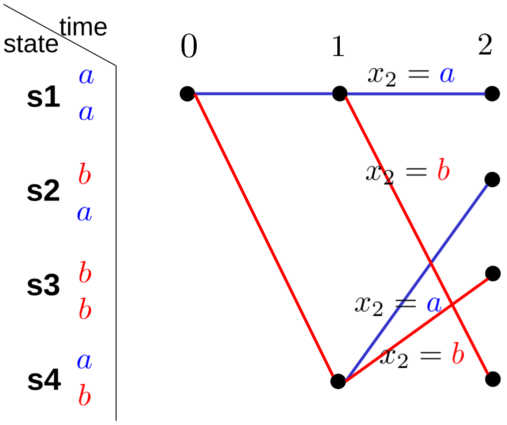


Figure 5: A trellis diagram at $i = 2$.

Cost calculation

Remember in the canonical optimization (4) that we are interested in the cost $c_i(s_i)$. So one natural question that arises is: How can we calculate the cost in (4) from the trellis diagram? To figure this out, consider a concrete example for $n = 4$; see Fig. 6. This is the example in which $(x_1, x_2, x_3, x_4) = (a, b, a, a)$, so the trellis path takes the blue-red-blue-blue transition arrows, yielding the state change as: s1–s1–s4–s2–s1. To ease cost calculation, here we leave an associated cost at the corresponding state node. For instance, we put $c_1([a; a])$ (marked in green in Fig. 6) nearby the black dot s1 in time 1. Similarly we leave $c_2([a; b])$, $c_3([b; a])$, $c_4([a; a])$ for the corresponding black dots. By aggregating all the costs associated with the black dots, we can readily compute the cost for one such sequence. Considering all possible sequence patterns,

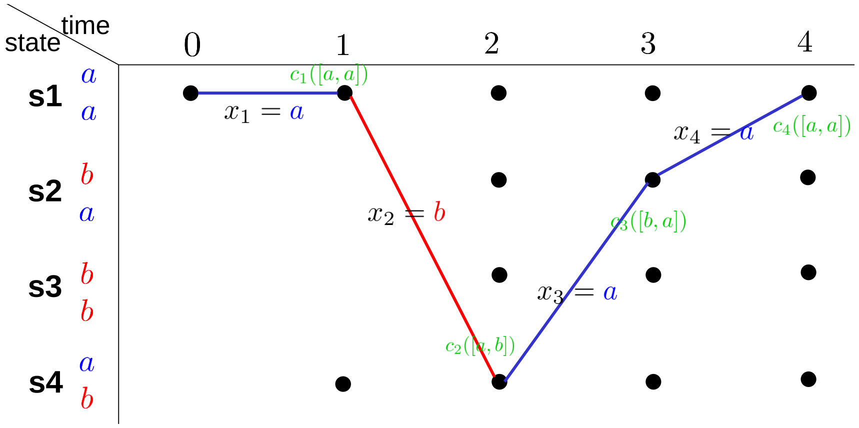


Figure 6: A trellis diagram for the sequence $(x_1, x_2, x_3, x_4) = (a, b, a, a)$.

we can compute the optimal path as:

$$\mathbf{s}^*_{\mathsf{MAP}} = \arg\min_{s_1,s_2,s_3,s_4} \{c_1(s_1) + c_2(s_2) + c_3(s_3) + c_4(s_4)\} .$$

As mentioned earlier, a naive *exhaustive* search requires the number 2^4 of cost calculations – so the complexity is very expensive especially for a large n.

Look ahead

Next time, we will study the Viterbi algorithm that well exploits the structure of the trellis diagram to find $\mathbf{s}^*$ efficiently.

EE210 Probability and Introductory Random Processes June 3, 2021
KAIST, Spring 2021 Changho Suh (chsuh@kaist.ac.kr)

Lecture 25: Speech recognition: Viterbi algorithm I

Recap

During the past two lectures, we showed that speech recognition is an inference problem and then derived the optimal MAP estimator. We also simplified the MAP estimator by exploring the nice statistical structure on $\{(X_i, Y_i)\}_{i=1}^n$. Here X_i indicates the ith phoneme and Y_i denotes the ith feature (spectral information). See Fig. 1. The MAP estimator reads:

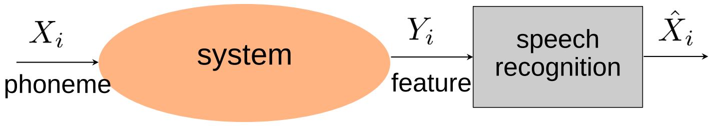


Figure 1: A block diagram of the speech recognition system and recovery block

$$\hat{\mathbf{x}}_{\mathsf{MAP}} = \arg \max_{x_1,\ldots,x_n \in \mathcal{X}} \mathbb{P}_{\mathbf{X}}(x_1,\ldots,x_n) f(y_1,\ldots,y_n | x_1,\ldots,x_n). \tag{1}$$

Assuming a single-memory Markov model for $\{X_i\}_{i=1}^n$, we simplified the a priori probability as:

$$\mathbb{P}_{\mathbf{X}}(x_1,\ldots,x_n) = \mathbb{P}(x_1) \prod_{i=2}^{n} \mathbb{P}(x_i | x_{i-1}).$$

Exploiting the HMM structure of $\{Y_i\}_{i=1}^n$, we could figure out that the likelihood is the product of every individual:

$$f(y_1,\ldots,y_n | x_1,\ldots,x_n) = \prod_{i=1}^{n} f(y_i | x_i).$$

Plugging the above two into (1), we then obtained the simplified MAP estimator:

$$\hat{\mathbf{x}}_{\mathsf{MAP}} = \arg \max_{x_1,\ldots,x_n \in \mathcal{X}} \mathbb{P}(x_1) \prod_{i=2}^{n} \mathbb{P}(x_i | x_{i-1}) \prod_{i=1}^{n} f(y_i | x_i). \tag{2}$$

One naive way to derive $\hat{\mathbf{x}}_{\mathsf{MAP}}$ is to search for all the possible sequence patterns that $\{x_i\}_{i=1}^n$ can take on. However, this way, called the *exhaustive search*, comes with a challenge. The challenge is that the total number of possible sequence patterns is huge, $|\mathcal{X}|^n = 44^n$. The complexity grows *exponentially* with n.

Today's lecture

For the remaining two lectures including today's one, we will study another very popular method that addresses the computational challenge. The method is called the *Viterbi algorithm*. Today we will investigate several concepts that form the basis of algorithm. Next time, we will figure out how the algorithm works in detail. This lecture consists of three parts. Actually the "Viterbi" is the last name of the inventor of the algorithm. First off, I will explain who the inventor is, and will emphasize the key feature of the algorithm w.r.t. complexity prior to diving into details. It turns out the Viterbi algorithm is a very *generic* algorithm with a wide variety of applications, yet being subject to a particular form of optimization. So in the second part, we will translate the original optimization (2) into another respecting the particular form. Lastly we will study three important concepts which serve to explain how the algorithm works.

The key feature of Viterbi algorithm

The inventor of the algorithm is Andrew Viterbi. See Fig. 2. He is one of the giants in the fields of communication and information theory, also known as a co-founder of Qualcomm Inc that you may hear of. Search for wikipedia. You will soon figure out he is indeed a giant figure. Actually he came up with the algorithm in the process of addressing a different yet interesting problem that arises in communication. Later, many people figured out the algorithm can be applicable to a widening array of problems beyond communication, including speech recognition of our current interest.

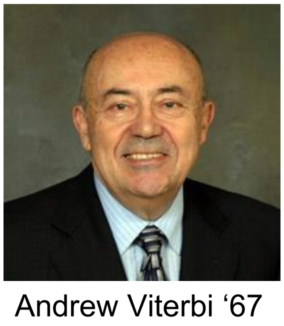


Figure 2: Andrew Viterbi is a giant in the communication and information theory fields.

The key feature of the algorithm that I want to emphasize is that the algorithm complexity grows *linearly* with n, being much smaller than 44^n that we faced with the exhaustive search. As you may figure out later, the algorithm has nothing to do with probability of this course's focus. As emphasized in the key feature, however, the algorithm is so powerful with extremely lower complexity. This is the sole reason that I want to introduce this algorithm to you guys. Please be familiar with the Viterbi algorithm in your entire career. The algorithm will constantly benefit you, as long as you are working on something relevant to optimization.

Translation into the canonical form

Now let's start by translating the original optimization (2) into another with the canonical structure that the algorithm relies upon. To this end, we first massage the optimization (2):

$$\hat{\mathsf{x}}_{\mathsf{MAP}} = \arg \max_{x_1,\ldots,x_n \in \mathcal{X}} \mathbb{P}(x_1) \prod_{i=2}^{n} \mathbb{P}(x_i|x_{i-1}) \prod_{i=1}^{n} f(y_i|x_i).$$

Setting $x_0 = 0$ (nothing), we can simplify the optimization as:

$$\begin{aligned}
\hat{\mathbf{x}}_{\mathsf{MAP}} &= \arg \max_{x_1,\ldots,x_n \in \mathcal{X}} \prod_{i=1}^{n} \mathbb{P}(x_i|x_{i-1}) f(y_i|x_i) \\
&= \arg \max_{x_1,\ldots,x_n \in \mathcal{X}} \sum_{i=1}^{n} \log \left\{ \mathbb{P}(x_i|x_{i-1}) f(y_i|x_i) \right\} \\
&= \arg \min_{x_1,\ldots,x_n \in \mathcal{X}} \sum_{i=1}^{n} \underbrace{\log \left\{ \frac{1}{\mathbb{P}(x_i|x_{i-1}) f(y_i|x_i)} \right\}}_{=:c_i(s_i)}
\end{aligned}$$

where the 2nd equality comes from the fact that taking an increasing function $\log(\cdot)$ does not alter the maximizer; and the last equality is due to changing the sign of the objective function. Notice that the ith component in the summation in the last line is a function of (x_{i-1}, x_i, y_i). So one can denote the ith component by $c_i(s_i)$ where

$$s_i := \begin{bmatrix} x_{i-1} \\ x_i \end{bmatrix}. \tag{3}$$

Here we call s_i the *state*. Later you will figure out the rationale behind the naming. The reason we leave the subscript i in c is that the ith component is also a function of y_i. Notice that decoding $(x_1, x_2, \ldots, x_n)$ is equivalent to decoding $(s_1, s_2, \ldots, s_n)$. Also remember that y_i's are given and x_i's are optimization variables. So the original optimization (2) is equivalent to:

$$\hat{\mathbf{s}}_{\mathsf{MAP}} = \arg \min_{s_1,\ldots,s_n} \sum_{i=1}^{n} c_i(s_i) \tag{4}$$

where

$$c_i(s_i) = \log \left\{ \frac{1}{\mathbb{P}(x_i|x_{i-1}) f(y_i|x_i)} \right\}. \tag{5}$$

The above optimization (4) is the canonical optimization that the Viterbi algorithm is built upon. We are now ready to study the three concepts that form the basis of the algorithm.

Concept #1: Cost

The first concept is concerned about the interested quantity $c_i(s_i)$ in the canonical optimization (4). Here the quantity $c_i(s_i)$ can be viewed as something *negative*, because the smaller quantity, the better the situation is. Hence, we call it "cost". Now you can see why I use the notation c to denote the quantity.

Concept #2: State s_i

The second concept is the state s_i that we defined in (3). The state has certain properties. For ease of explanation of the properties, let us assume that x_i takes one among only two possible phonemes, say $/a/$ and $/b/$. For notational simplicity, let us drop the slash symbol: $x_i \in \{a, b\}$. In this case, we can readily see that s_i can take one of the following six candidates:

$$\begin{bmatrix} 0 \\ b \end{bmatrix}, \begin{bmatrix} 0 \\ a \end{bmatrix}, \underbrace{\begin{bmatrix} a \\ a \end{bmatrix}}_{s1}, \underbrace{\begin{bmatrix} b \\ a \end{bmatrix}}_{s2}, \underbrace{\begin{bmatrix} b \\ b \end{bmatrix}}_{s3}, \underbrace{\begin{bmatrix} a \\ b \end{bmatrix}}_{s4}. \tag{6}$$

Observe that the first two occur only at the beginning ($i = 1$), assuming that $x_0 = 0$. So the two states are negligible relative to the others for a large value of n. For simplification, let us not worry about the two states. To this end, we intentionally set $x_0 = a$. In this case, we have only four possible states. The number "four" is obviously finite. There is a terminology which indicates an entity w.r.t. such state that can take one among only the *finite* number of possible candidates. That is, a *Finite State Machine*, FSM for short.

The FSM has an interesting property which can be illustrated in picture. To see this, first observe that each state s_i can move from one to another, depending on the value of x_{i+1}. So one can now think of the following state translations as illustrated in Fig. 3. Here a or b labeled above a transition arrow indicates the value of x_{i+1}, given the current state s_i. For instance, suppose that $x_{i+1} = b$ given the current state s1. Then, we move along the red transition arrow to arrive at the state s4 of $[a; b]$. The picture like Fig. 3 is called the *state transition diagram*.

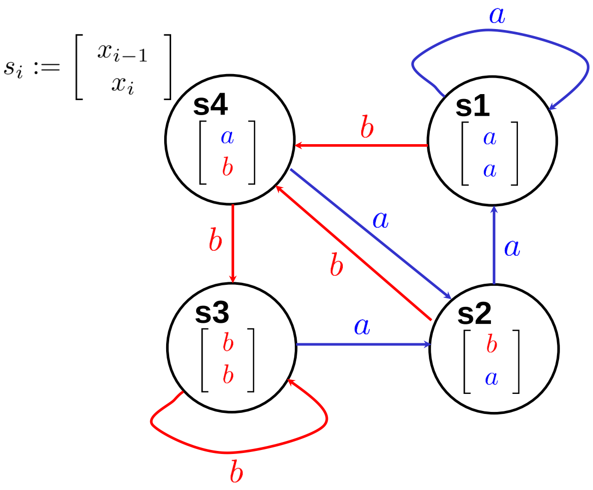

$s_i := \begin{bmatrix} x_{i-1} \\ x_i \end{bmatrix}$
s4
$\begin{bmatrix} a \\ b \end{bmatrix}$
s1
$\begin{bmatrix} a \\ a \end{bmatrix}$
s3
$\begin{bmatrix} b \\ b \end{bmatrix}$
s2
$\begin{bmatrix} b \\ a \end{bmatrix}$
a
b
a
b
b
a
a
b

Figure 3: A finite state machine with four states. A transition occurs depending on the value of x_{i+1}.

While the state transition diagram well represents how each state moves around to another, it does not capture the *time evolution*. This is exactly where another concept w.r.t. the FSM arises. That is, the *trellis diagram*. Below we will describe how the trellis diagram looks like and then will explain how it helps solve the interested optimization (4) that concerns the given information $\{y_i\}_{i=1}^n$.

Concept #3: Trellis diagram

The trellis diagram exhibits both state transitions and time evolution. To clearly understand how it works, let me walk you through details with the help of Figs. 4 and 5.

Consider the state at time 1:

$$s_1 = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix} = \begin{bmatrix} a \\ x_1 \end{bmatrix} = \begin{cases} \begin{bmatrix} a \\ a \end{bmatrix} = \text{s1}, & \text{if } x_1 = a; \\ \begin{bmatrix} a \\ b \end{bmatrix} = \text{s4}, & \text{if } x_1 = b, \end{cases}$$

where the second equality is due to our assumption $x_0 = a$. Notice that s_1 takes one of the two possible states s1 and s4, reflected in the two dots at time 1 in Fig. 4. In time 0, we have two options for s_0 depending on the value of x_{-1}: s1 and s2. To remove the ambiguity in time 0, let

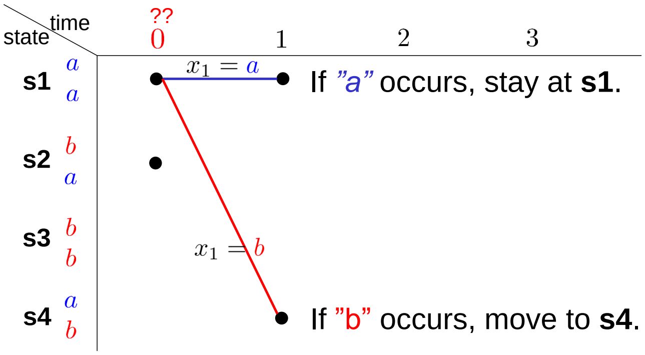


Figure 4: A trellis diagram at $i = 1$.

us further assume $x_{-1} = a$. This way, we can fix the initial state as s1. So we ensure that we start always from s1. As mentioned earlier, if $x_1 = a$, then $s_1 = \text{s1}$; otherwise $s_1 = \text{s4}$.

To be more familiar with how it works, let us consider one more time slot as illustrated in Fig. 5. Suppose $s_1 = \text{s1}$. If $x_2 = a$, then it stays at the same state s1 (reflected in the blue transition arrow); otherwise, it moves to s4 (reflected in the red arrow). On the other hand, given $s_1 = \text{s4}$, if $x_2 = a$, then it moves to s2; otherwise, it goes to s3.

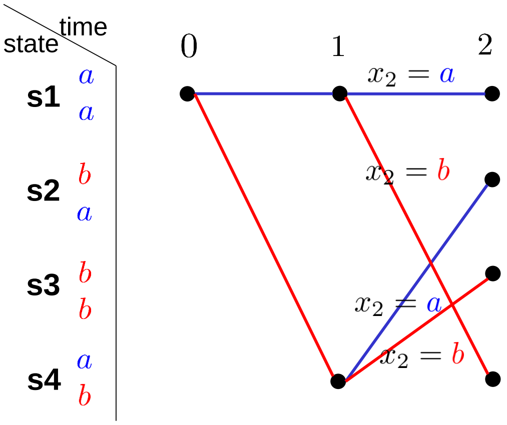


Figure 5: A trellis diagram at $i = 2$.

Cost calculation

Remember in the canonical optimization (4) that we are interested in the cost $c_i(s_i)$. So one natural question that arises is: How can we calculate the cost in (4) from the trellis diagram? To figure this out, consider a concrete example for $n = 4$; see Fig. 6. This is the example in which $(x_1, x_2, x_3, x_4) = (a, b, a, a)$, so the trellis path takes the blue-red-blue-blue transition arrows, yielding the state change as: s1–s1–s4–s2–s1. To ease cost calculation, here we leave an associated cost at the corresponding state node. For instance, we put $c_1([a; a])$ (marked in green in Fig. 6) nearby the black dot s1 in time 1. Similarly we leave $c_2([a; b])$, $c_3([b; a])$, $c_4([a; a])$ for the corresponding black dots. By aggregating all the costs associated with the black dots, we can readily compute the cost for one such sequence. Considering all possible sequence patterns,

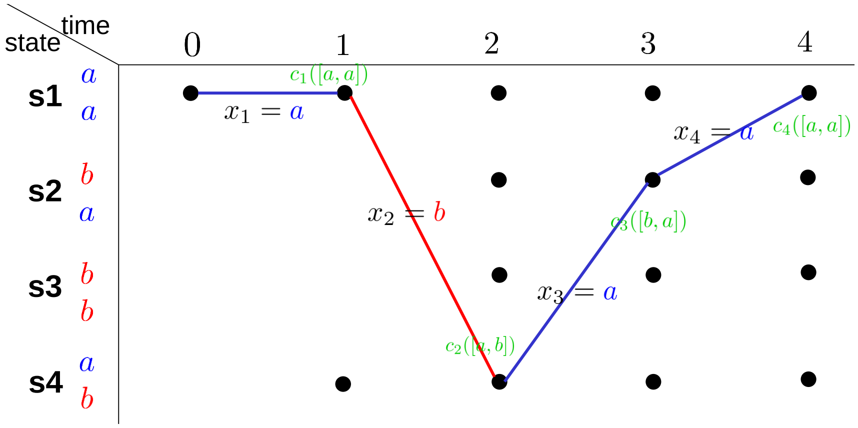


Figure 6: A trellis diagram for the sequence $(x_1, x_2, x_3, x_4) = (a, b, a, a)$.

we can compute the optimal path as:

$$\mathbf{s}^*_{\mathsf{MAP}} = \arg \min_{s_1,s_2,s_3,s_4} \{c_1(s_1) + c_2(s_2) + c_3(s_3) + c_4(s_4)\} .$$

As mentioned earlier, a naive *exhaustive* search requires the number 2^4 of cost calculations – so the complexity is very expensive especially for a large n.

Look ahead

Next time, we will study the Viterbi algorithm that well exploits the structure of the trellis diagram to find $\mathbf{s}^*$ efficiently.

EE210 Probability and Introductory Random Processes June 8, 2021
KAIST, Spring 2021 Changho Suh (chsuh@kaist.ac.kr)

Lecture 26: Speech recognition: Viterbi algorithm II

Recap

Last time, we have studied three concepts which I claimed serve to explain the mechanism of the Viterbi algorithm intended for computing the optimal MAP estimator:

$$\hat{\mathsf{s}}_{\mathsf{MAP}} = \arg \min_{s_1,\ldots,s_n} \sum_{i=1}^{n} c_i(s_i) \tag{1}$$

where

$$c_i(s_i) := \log \left\{ \frac{1}{\mathbb{P}(x_i|x_{i-1}) f(y_i|x_i)} \right\} \quad \text{and} \quad s_i := \begin{bmatrix} x_{i-1} \\ x_i \end{bmatrix}. \tag{2}$$

The first concept is the cost $c_i(s_i)$ that represents an interested quantity in the canonical optimization (1). The second is the state s_i concerning a finite state machine having the following states:

$$\underbrace{\begin{bmatrix} a \\ a \end{bmatrix}}_{s1}, \underbrace{\begin{bmatrix} b \\ a \end{bmatrix}}_{s2}, \underbrace{\begin{bmatrix} b \\ b \end{bmatrix}}_{s3}, \underbrace{\begin{bmatrix} a \\ b \end{bmatrix}}_{s4}.$$

The third is the trellis diagram that visualizes how each state changes in time. Fig. 1 illustrates an example of the trellis diagram for the sequence $(x_1, x_2, x_3, x_4) = (a, b, a, a)$, assuming that $x_0 = x_{-1} = a$.

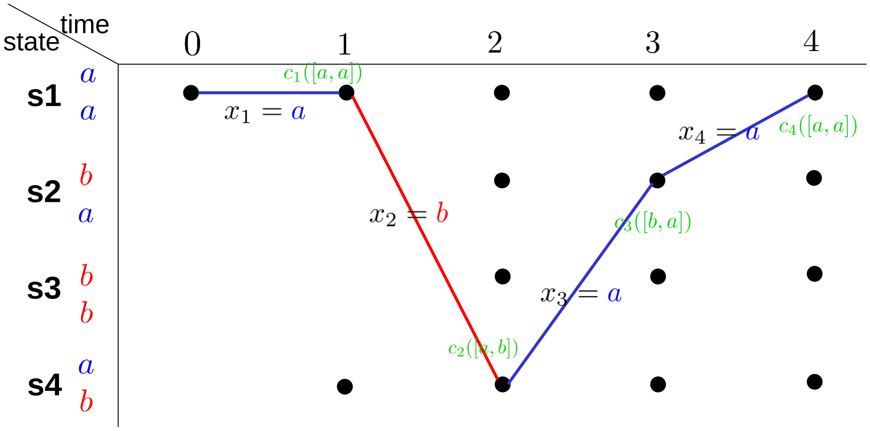


Figure 1: A trellis diagram for the sequence $(x_1, x_2, x_3, x_4) = (a, b, a, a)$.

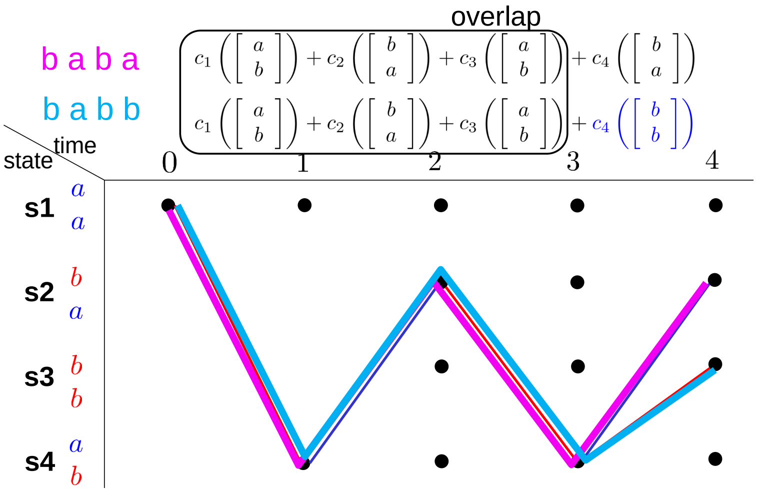
Today's lecture

Today we will study how the Viterbi algorithm works in detail. Specifically what we are going to do are three-folded. I will first emphasize one key observation which gives a significant insight into the algorithm. We will then study how the algorithm works. Lastly we will show that the computational complexity grows indeed *linearly* with n, as I claimed earlier. Since today is the last day for this course, I will also leave a few closing remarks in the end.

Key observation

The Viterbi algorithm is inspired by the following key observation. To see this clearly, let us consider two possible sequence patterns presented in Fig. 2:

$$\text{(i) } (x_1, x_2, x_3, x_4) = (b, a, b, a) \text{ (marked in purple);}$$
$$\text{(ii) } (x_1, x_2, x_3, x_4) = (b, a, b, b) \text{ (marked in blue).}$$



overlap
b a b a
$c_1\left(\begin{bmatrix} a \\ b \end{bmatrix}\right) + c_2\left(\begin{bmatrix} b \\ a \end{bmatrix}\right) + c_3\left(\begin{bmatrix} a \\ b \end{bmatrix}\right) + c_4\left(\begin{bmatrix} b \\ a \end{bmatrix}\right)$
b a b b
$c_1\left(\begin{bmatrix} a \\ b \end{bmatrix}\right) + c_2\left(\begin{bmatrix} b \\ a \end{bmatrix}\right) + c_3\left(\begin{bmatrix} a \\ b \end{bmatrix}\right) + c_4\left(\begin{bmatrix} b \\ b \end{bmatrix}\right)$
time
state
0
1
2
3
4
s1 a a
s2 b a
s3 b b
s4 a b

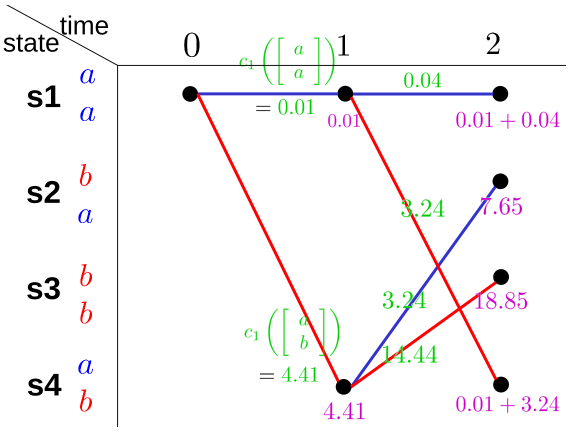
Figure 2: Key observation.

Here the key observation is that the two trellis paths are *significantly overlapped*; hence, the two corresponding costs are identical except for the cost w.r.t. the last state vector. This motivated Viterbi to come up with the following natural idea.

Idea of the Viterbi algorithm

The idea is to successively store only an *aggregated cost* up to time t and then use this to compute a follow-up aggregated cost w.r.t. the next time slot. In order to understand what this means in detail, let us consider a simple example. See Figs. 3 and 4.

For illustrative purpose, let us consider a simple setting ($n = 4$) in which cost computations are



state
time
0
1
2
s1
a
a
s2
b
a
s3
b
b
s4
a
b
$c_1\left(\begin{bmatrix}a\\a\end{bmatrix}\right)$
= 0.01
0.01
0.04
0.01 + 0.04
3.24
7.65
3.24
18.85
14.44
$c_1\left(\begin{bmatrix}a\\b\end{bmatrix}\right)$
= 4.41
4.41
0.01 + 3.24

Figure 3: Cost computation and store strategy.

already done via (2) for all the possible states and time slots:

$$\begin{bmatrix} c_1\left(\begin{bmatrix}a\\a\end{bmatrix}\right)\\ c_1\left(\begin{bmatrix}b\\a\end{bmatrix}\right)\\ c_1\left(\begin{bmatrix}b\\b\end{bmatrix}\right)\\ c_1\left(\begin{bmatrix}a\\b\end{bmatrix}\right) \end{bmatrix} = \begin{bmatrix}0.01\\ *\\ *\\ 4.41\end{bmatrix}, \quad \begin{bmatrix} c_2\left(\begin{bmatrix}a\\a\end{bmatrix}\right)\\ c_2\left(\begin{bmatrix}b\\a\end{bmatrix}\right)\\ c_2\left(\begin{bmatrix}b\\b\end{bmatrix}\right)\\ c_2\left(\begin{bmatrix}a\\b\end{bmatrix}\right) \end{bmatrix} = \begin{bmatrix}0.04\\ 3.24\\ 14.44\\ 3.24\end{bmatrix},$$

$$\begin{bmatrix} c_3\left(\begin{bmatrix}a\\a\end{bmatrix}\right)\\ c_3\left(\begin{bmatrix}b\\a\end{bmatrix}\right)\\ c_3\left(\begin{bmatrix}b\\b\end{bmatrix}\right)\\ c_3\left(\begin{bmatrix}a\\b\end{bmatrix}\right) \end{bmatrix} = \begin{bmatrix}2.25\\ 0.25\\ 6.25\\ 0.25\end{bmatrix}, \quad \begin{bmatrix} c_4\left(\begin{bmatrix}a\\a\end{bmatrix}\right)\\ c_4\left(\begin{bmatrix}b\\a\end{bmatrix}\right)\\ c_4\left(\begin{bmatrix}b\\b\end{bmatrix}\right)\\ c_4\left(\begin{bmatrix}a\\b\end{bmatrix}\right) \end{bmatrix} = \begin{bmatrix}17.64\\ 4.84\\ 0.04\\ 4.84\end{bmatrix}$$

where $*$ means that cost computations are omitted since they are not needed in the beginning.

The cost $c_1([a;a]) = 0.01$ is placed along the blue transition arrow, as illustrated in Fig. 3. We then store the cost at the state s1 node at time 1 (a black dot). Here we indicate the store by marking a purple-colored number nearby the black dot. We do the same thing w.r.t. the cost yet now when $x_1 = b$. We place the cost $c_1([a;b]) = 4.41$ along the associated red transition arrow and then store 4.41 nearby the black dot w.r.t. the state s4.

Next we consider the cost $c_2([a;a]) = 0.04$. So the aggregated cost up to time 2 w.r.t. the state s1 would be 0.01+0.04, as illustrated in Fig. 3. Similarly the aggregated costs for the other states (s2, s3, s4) would be 4.41+3.24, 4.41+ 14.44, 0.01+3.24, respectively.

Now one can see the core idea of the Viterbi algorithm from time 3. See Fig. 4. Consider the state s1 at time 3. This state occurs when $x_3 = a$ and comes from two possible prior states: s1 and s2. We consider the cost $c_3([a;a]) = 2.25$. So the aggregated cost assuming that it comes from the prior s1 state (storing 0.05 for the aggregated cost up to time 2) would be: $0.05 + 2.25 = 2.3$. On the other hand, the aggregated cost w.r.t. the prior s2 state would be: $7.65 + 2.25 = 9.9$. Now how to deal with the two cost values? Remember at the end of the day that we are interested in

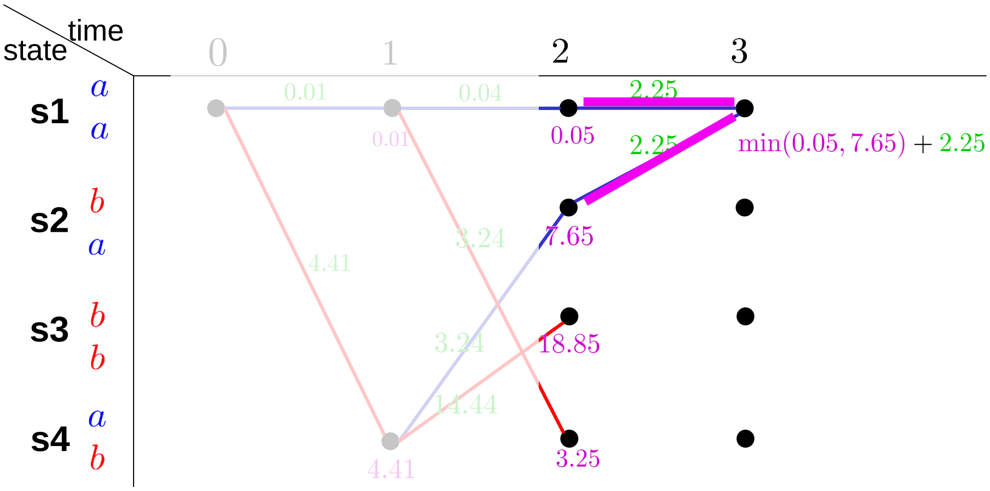


Figure 4: Ided of the Viterbi algorithm

finding the path that yields the *minimum* aggregated cost. So the path w.r.t. the larger cost 9.9 would be eliminated in the competition. So we don't need to worry about any upcoming paths w.r.t. the larger cost. This naturally motivates us to store only the *minimum* between the two cost values at the state s1 in time 3, while ignoring the other loser path. So we store 2.3 at the node, as illustrated in Fig. 5. We do the same thing for the other states. For the state s2, the

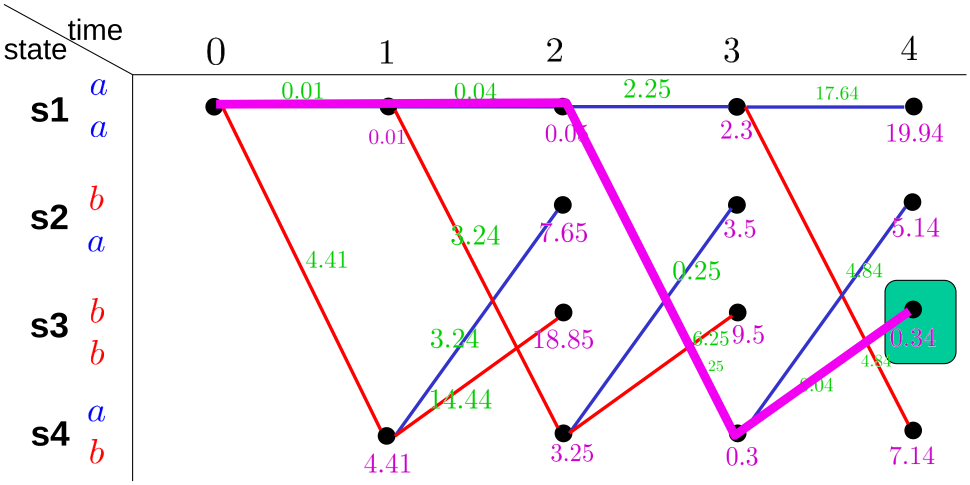


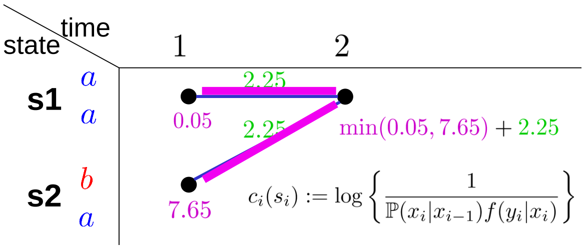
Figure 5: Choose $\mathbf{s}^*$ that minimizes the aggregated cost.

lower path turns out to be the winner, so we store the corresponding aggregated cost 3.5 at the state s2 node, while deleting the upper loser path. Similarly for the states s3 and s4. We repeat this procedure until the last time slot. See all the associated computations in Fig. 5.

Now how to find the path that yields the minimum aggregated cost from the picture? It is very simple. Take a look at the four aggregated costs at the last time slot: 19.94, 5.14, 0.34, 7.14. We then pick up the minimum cost 0.34. Now how to find the corresponding sequence pattern? Since we leave only the survivor paths in the picture while deleting loser paths, we can readily find the path via *backtracking*. The survivor paths form the thick purple trajectory in Fig. 5, which corresponds to the sequence $(x_1, x_2, x_3, x_4) = (a, a, b, b)$.

Complexity

Remember I argued that the complexity of the Viterbi algorithm grows linearly with n. To see this clearly, let us consider the essential operation that occurs in each node. Focus on the operation w.r.t. the state s1 node at time 3, as illustrated in Fig. 6. Given the computations of the a prior probability and the likelihood function (which can be stored in a table), the cost computation requires: 1 log operation; 1 multiplication; 1 addition; and 1 comparison. The comparison operation is needed for taking the minimum between the two. This operation is repeated for the other states spanning the entire time slots, reflected in all the dots in the picture. Since the total number of black dots is $4n$, the complexity of the Viterbi algorithm grows linearly with n.



time
state
1
2
s1
a
a
s2
b
a
2.25
0.05
2.25
$\min(0.05, 7.65) + 2.25$
7.65
$c_i(s_i) := \log\left\{\frac{1}{\mathbb{P}(x_i|x_{i-1})f(y_i|x_i)}\right\}$

Figure 6: Complexity per state.

Closing

Finally I would like to leave a remark which may be helpful for your future careers. One key message that I would like to deliver from this course is that fundamental concepts and principles (probability concepts and MAP/ML principles) play important roles. So in view of this, I strongly recommend you to make every efforts for *being strong at fundamentals!* The fundamentals that I would like to put a special emphasis on are w.r.t. modern AI technologies that you may be very interested in. As you may image, being an expert in the AI field requires many backgrounds. One major background that I believe crucial is *mathematics*, in particular four fundamental branches in mathematics.

The first is *optimization*. Remember that the goal of machine learning can be achieved through optimization. The second is a field which provides instrumental tools with which one can translate the objective function and/or constraints into simple and tractable formulas. The field is: *linear algebra*. As you may be familiar with, many seemingly-complicated mathematical formulas can be expressed as simple terms that involve matrix multiplications and additions. The third is a field that plays a role in dealing with uncertainty that appears in random quantities. The field is: *probability*. The last is a field that serves to shed optimal architectural insights into machine learning models of interest. That is: *information theory*. Remember the role of cross entropy (an information-theoretic notion) in the design of the optimal loss function.

These are the very fundamentals which I believe are crucial for advancing the 4th industrial revolution empowered by AI technologies. So my advice is: Be strong at these fundamentals. Here are some relevant courses offered at KAIST:

1. EE424: Introduction to Optimization;
2. MA109: Introduction to Linear Algebra;

3. EE210: Introduction to Probability & Random Processes;
4. EE326: Introduction to Information Theory & Coding.

One caveat here is that such fundamentals are highly likely to be built only when you are at school. Of course it is a bit exaggerated, but it seems indeed the case according to the experiences of my own and many others. You may be able to understand what this means *after* you graduate; not enough time would be given for you to deeply understand some principles and also your stamina would not be as good as that of now.