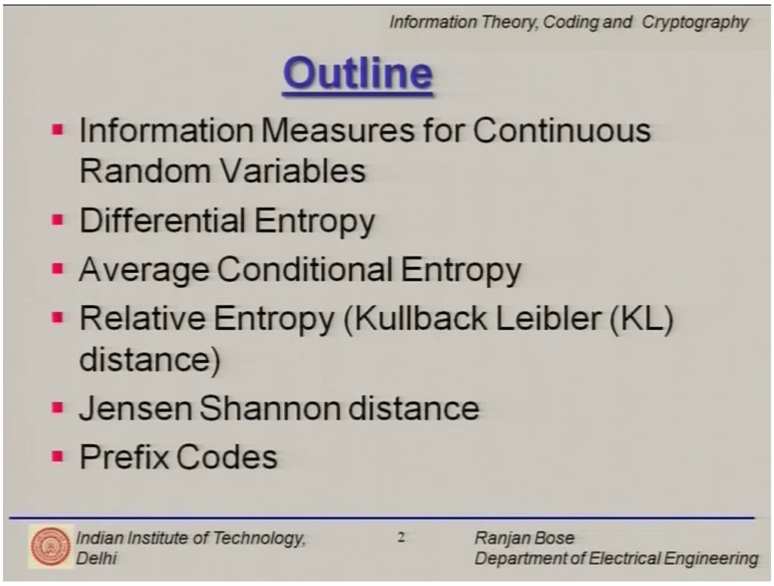
Information Theory, Coding and Cryptography
Dr. Ranjan Bose
Department of Electrical Engineering
Indian Institute of Technology, Delhi

Module - 03
Source Coding
Lecture – 03

(Refer Slide Time: 00:24)

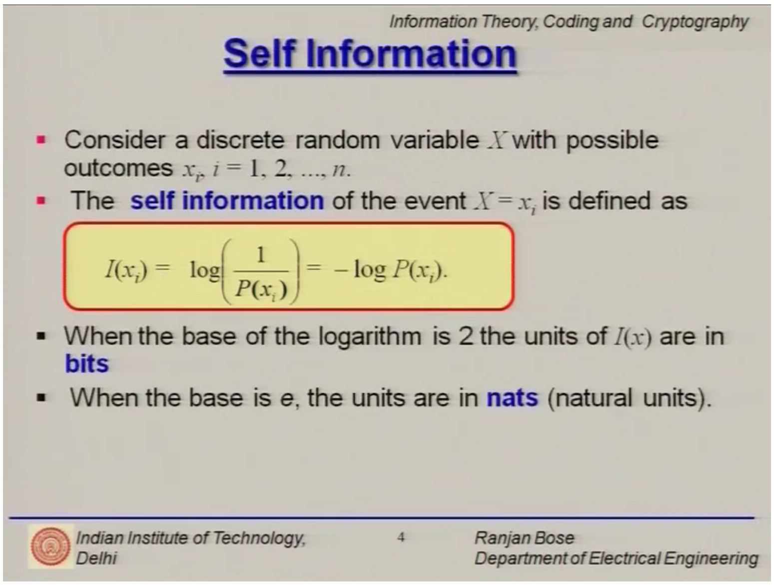


Hello and welcome to module 3 of source coding. Let us look at the outline of today's talk. Today we are going to consider a very interesting thing, information measures for continuous random variables. So, far we have only dealt with discrete random variables. So, what does it mean? We will look at that. Then we will formulate the notion of differential entropy, we will follow it up with Average Conditional Entropy. We look at Relative Entropy which is also known as Kullback Leibler Distance. We will look at Jensen Shannon's distance and finally, we will look at Prefix Codes ok.

Let us start once again. We will cut off the first three minutes all right. All of you are settled in ok, shall start with the regular formality hello and welcome to module three of source coding. Let us look at the outline of today's talk. We will start with information measures for continuous random variable as opposed to discrete random variables that we talked about in the previous module. Then we will formulate the notion of Differential Entropy. We will look at Average Conditional Entropy for Continuous

Random Variables. Then we will discuss something called Relative Entropy which is a kind of distance measure called the Kullback Leibler Distance. We will then look at Jensen Shannon's distance and finally, we will introduce the notion of Prefix Codes. So, this is our general outline. But first let us start with a quick recap of what we have done already.

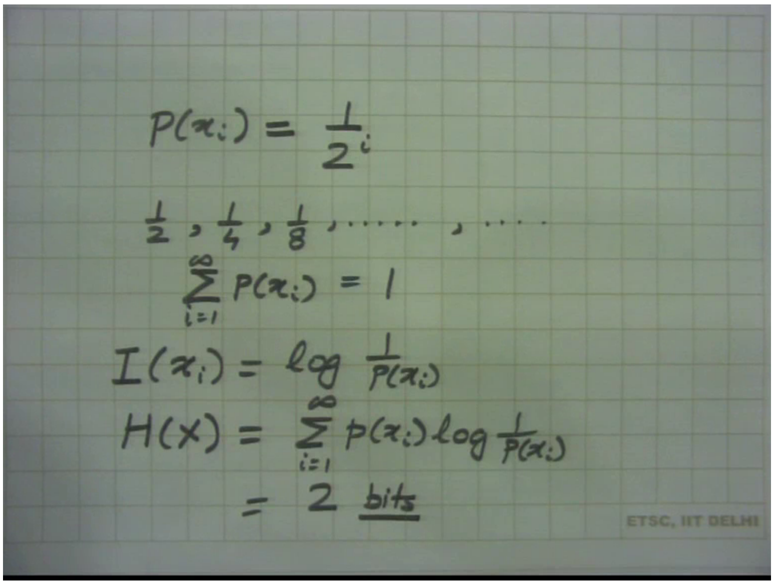
(Refer Slide Time: 02:38)



So, we have already looked at Average Mutual Information. We talked about entropy and its relation to say self-information. We went on to discuss conditional entropy, joint entropy and so and so forth.

So, very quick look at what self information was. So, if you remember we talked about a discrete random variable X with possible outcomes x i equal to 1, 2, 3 up to n and self information was defined as I x i log P 1 over x i and when the base of the log was 2, the units was in bits, but please note this n does not have to be finite; what if i goes from 1, 2, 3, 4 up to n. Let us look at an example.

(Refer Slide Time: 03:41)

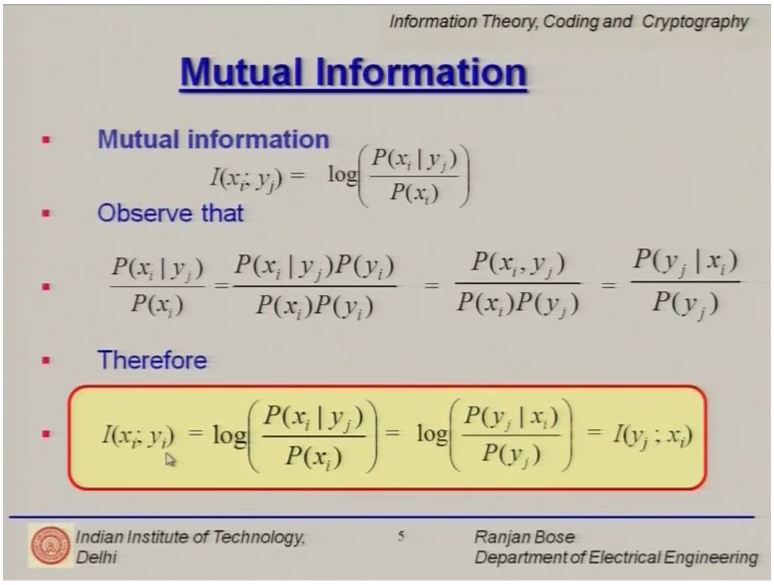


So, suppose I have P x i equal to 1 over 2 raise power i right. So, the probabilities look like 1 by 2, 1 by 4, 1 by 8 and so and so forth, but I do not stop at n. I go right up to infinity. Now if you do a basic sanity check overall I, you will see that it adds up to 1. So, this is indeed a probability measure and if it is a probability measure, then I should be able to talk about the self-information and what do I do? I plug into the formula. So, if I want to do I x i, I will have for a particular case log 1 over P x i, but if I want to have the notion of H of x which is the average self-information, then I have summation of P x i log 1 over P x i and this i will go from 1 to infinity.

Now, if I just plug in the values of the probabilities here and I solve it you can do so. It is a pretty straightforward answer because log 1 over 2 raise power 1 2 raise power 2 and so and so forth with probabilities multiplied here you will get up a summation and it adds up to 2 bits.

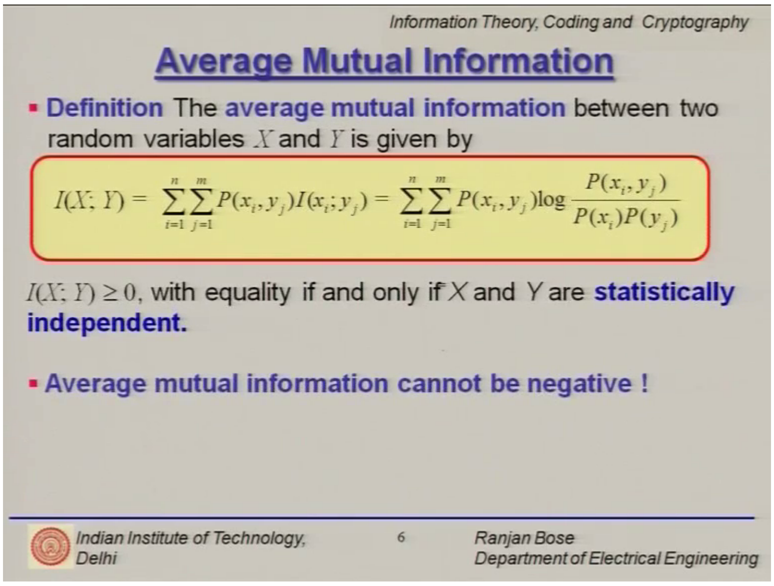
So, please note that even though there are infinite number of possible outcomes here, the net average self-information is bounded. So, even though the formula does not allow you to some beyond n, you can always have a answer up to a summation up to infinity.

(Refer Slide Time: 05:42)



We then looked at mutual information and mutual information between x i and y j was defined as log P x i given y j divided by P xi and we made a very interesting observation that I x i semicolon y j is equal to i y j semicolon x i. So, it is symmetric in nature.

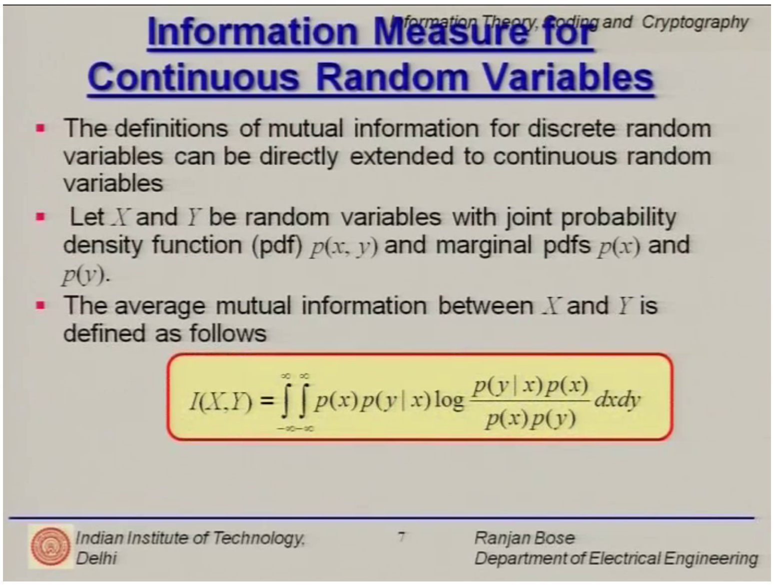
(Refer Slide Time: 06:05)



We then went on to define the notion of a average mutual information which is just a average it over the joint probabilities of P x i comma y j and I get capital I, the mutual information X semicolon Y is defined as follows and we know that I X semicolon Y is non-negative greater than or equal to 0 and the equality is achieved, if and only if X and

Y are statistically independent. Please note, these are discrete random variables. What we have to do today is to look at continuous random variables and whether there is a notion of mutual information for continuous random variables or not.

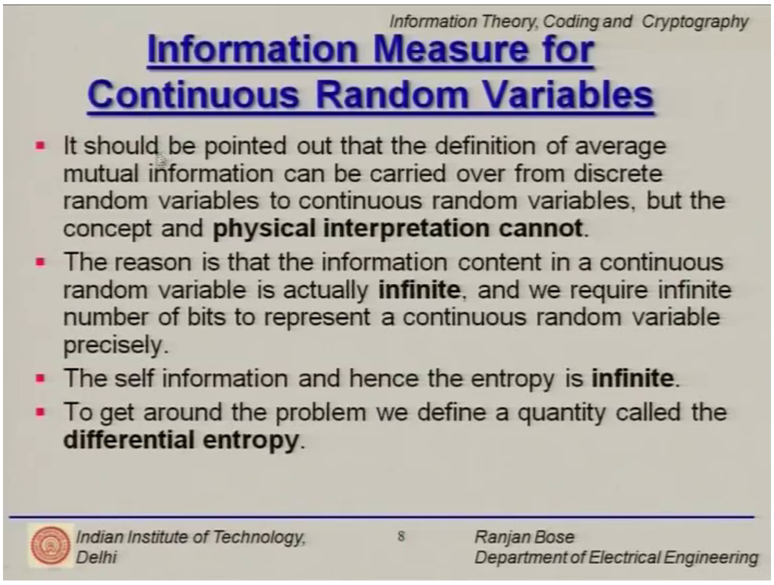
(Refer Slide Time: 06:56)



So, let us look at this notion of information measure for continuous random variables. Now the definition of mutual information for discrete random variables can directly be extended to continuous random variables, but this is talking about the definition. We will talk about the meaning, whether the meaning can be extended or not in a later slide.

So, let X and b be random variables with joint probability density functions p x comma y and marginal pdf's p x and p y, fair enough? We are talking about continuous random variables here. And we then define the average mutual information between x and y as I x comma y is equal to integration double integration over this is p x comma y and log p y given x into p x over p x into p y dxdy. So, this is the definition of continuous random variable the information measure for that.

(Refer Slide Time: 08:00)



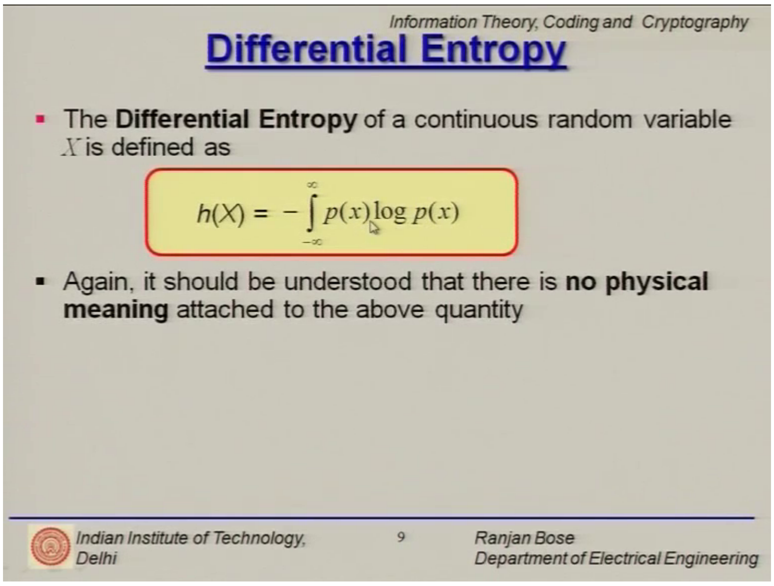
So, let us put a word of caution. Even though, we could extend the definition; the physical interpretation probably cannot be stretched. So, we should point out that the definition of average mutual information can be carried over from discrete random variables to continuous random variables, but the concept and the physical interpretation cannot.

What was the physical interpretation for a discrete random variable? Well, when we defined an average mutual information between X and Y, the basic physical interpretation was having observed Y, what can you say about X? In general on an average, what can you say? Maybe you can say something? Maybe you can say nothing. So, X occurrence of X communicate something about occurrence of Y and vice versa and that is basically captured by I X semicolon Y.

Now, you would like to say the same thing about continuous random variable, but unfortunately that is not the case. The reason is that the information contained in a continuous random variable is actually infinite. I mean what is the best way to look at it? Take a sample, represent it correctly. How many decimal points do you need? You can sample it and say it as 2.309921729, but you keep going. It is a continuous random variable. It is a point on the real line. So, you keep going and you really need infinite number of bits even to represent a single sample value, let alone the entire function.

So, the information content truly is infinite and therefore, we cannot really go on dealing with infinite information all the time. Let alone compare and what one communicates about the other random variable. So, we have just seen that the self-information entropy is infinite and we have to get around this problem and we define a new quantity called differential entropy. So, what each one of the random variables encompass infinite information? What about the difference? Maybe the difference is not infinite.

(Refer Slide Time: 10:47)



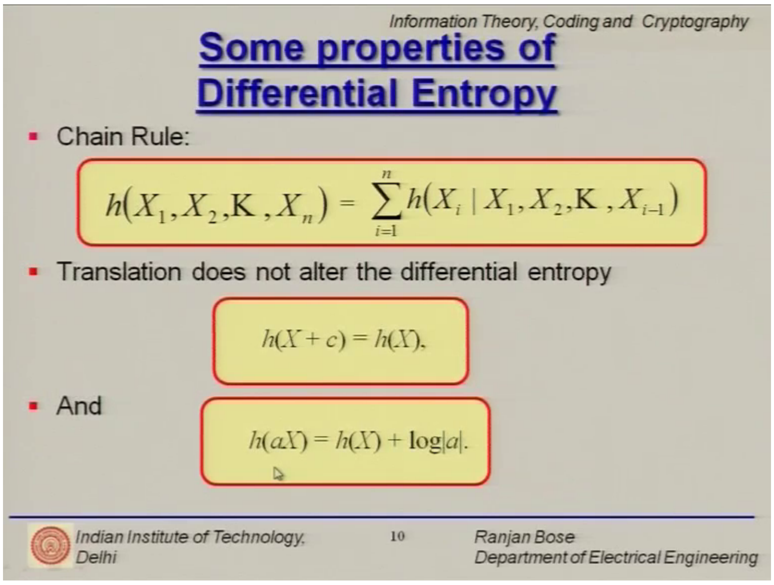
So let us define the differential entropy for a continuous random variable. So, the differential entropy is defined as h of X, mind you h is small h lowercase h as opposed to uppercase h for discrete random variables and is defined as an integration minus infinity to infinity p x log p x with a negative sign; as usual if the base of the log is 2, then the units are in bits. Same word of caution, there is no physical meaning attached to it.

If you remember in the earlier lectures, we tossed a coin and if it was a fair coin, we said that the average self-information for that source tossing a fair coin was 1 bit and it made sense because you needed 1 bit to represent either head or a tail. So, there is a strong physical interpretation; however, for a continuous random variable, we have no such luck.

Student: (Refer Time: 11:54).

Yes we will talk about this differential part very shortly, it will come.

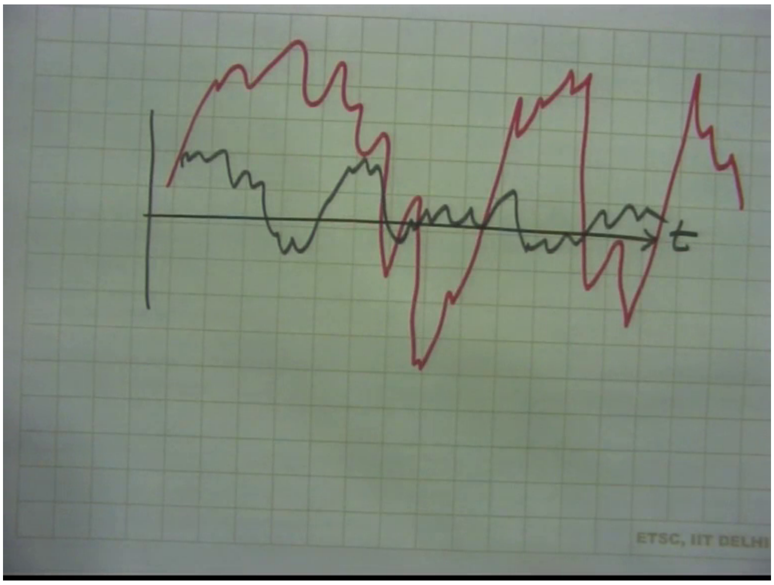
(Refer Slide Time: 12:00)



Now let us look at the properties of differential entropy. So, we talk about this chain rule where h X 1, X 2; this K should be dot, dot, dot, dot up to X n and it can be represented in a conditional form as h X i given X 1, X 2 dot, dot, dot, dot up to X i minus 1. The other property is that the translation of the random variable X does not change the differential entropy and I am relieved because if I add a constant, it really does not add to the randomness of X and the information content is primarily the measure for randomness.

So, translation it turns out; does not alter the differential entropy. And if you just multiply by a scalar, so the differential entropy of not X, but a times X is just a d c shift. How do you visualize this practically? Imagine a continuous random variable. So, let us plot this and try to get a physical interpretation for this.

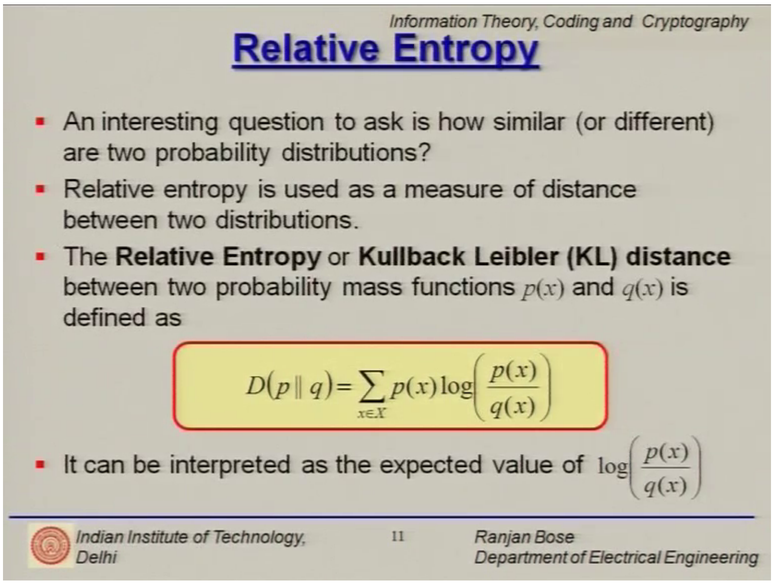
(Refer Slide Time: 13:36)



Suppose, this is just one capture of my random process because I have put a time axis here and suppose I pass it through an amplifier and multiply it with a. And now I get another capture of the same thing ok. So, if you can see just the physical observation tells you that the second random variable which is a times multiplied with the first one has a higher level of randomness.

The variance has gone up and consequently the information content has to go up, but how does it go up? It gives you a basic dc shift here. So, if you look at h a X equal to h of X plus log absolute value of a is a. There is a strong interpretation attached to the scaling factor.

(Refer Slide Time: 14:54)



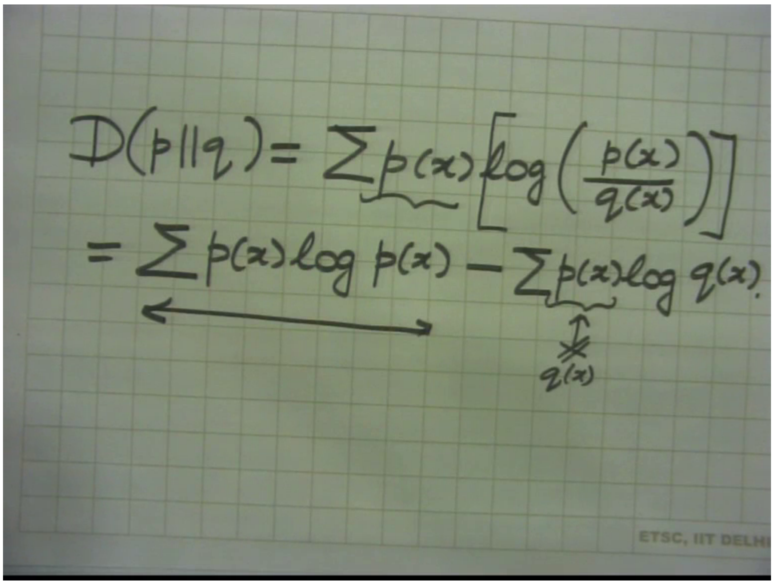
Now, another interesting question to ask is, how similar or if you are a pessimist; how different are two probability distributions fine? So, we talk about a notion of relative entropy as a measure of distance between two distributions, distance measure. Well I tell you, how different to probability distributions can be. If I ask you a question, I give you Gaussian and I say, how different is it with respect to another Gaussian or how different is a distribution with respect to a Rayleigh distribution? These questions are valid and we would like to have an answer to that.

So, we talk about this notion of relative entropy or by the two guys who defined Kullback and Leibler. This is called the Kullback Leibler distance between two probability mass functions; p x and q x. It is defined as d p parallel q is nothing, but p x log p x by q x ok. This you can see is nothing, but an expected value of log p x over q x. So, what they have done is taken the two quantities p x and q x, if you just think hard enough it is nothing, but p x log p x minus p x log q x, but if you remember p x log p x is a measure of the average self-information. So, therefore, the notion of relative entropy, how is ones information relative to the other?

Now.

Student: Sir, so what is the physical significance of p x log p x log x? It is actually a p x log p x there is a self-information and minus p x log p x.

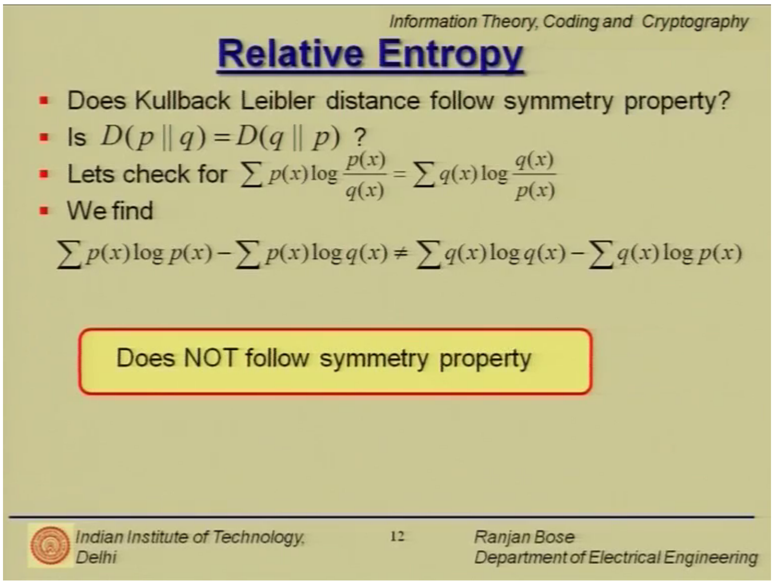
(Refer Slide Time: 17:21)



Yes. So, let us repeat the question. The question being asked is we have defined the Kullback Leibler distance as follows and with a little bit of imagination, you can see that this is nothing, but fair enough. Now, this has a physical interpretation. It tells me something about the average self-information of p of x; however, with a little bit of distortion because this is not q of x, otherwise I was really finding out the difference between the self-information, average self-information of p x and q x. So, they have kind of distorted it with a purpose and we will look at why it tells you because I am looking at a similarity measure between the two.

Now, why would they define it like this? It is very simple. Suppose p x and q x are identical, then you have log of 1 and clearly it is 0 and we are relieved to find the two distributions which are identical, their distance is 0. The more different they are, the larger should be the value of this distance and this notion; so, this is the basic definition. Logarithmic measure is required because right from the beginning, we have argued by a log measure for information is the only logical way to go and they have found out the difference between log p x and log q of x, but that just would not do. So, they have averaged it to give you the relative information measure.

(Refer Slide Time: 19:37)

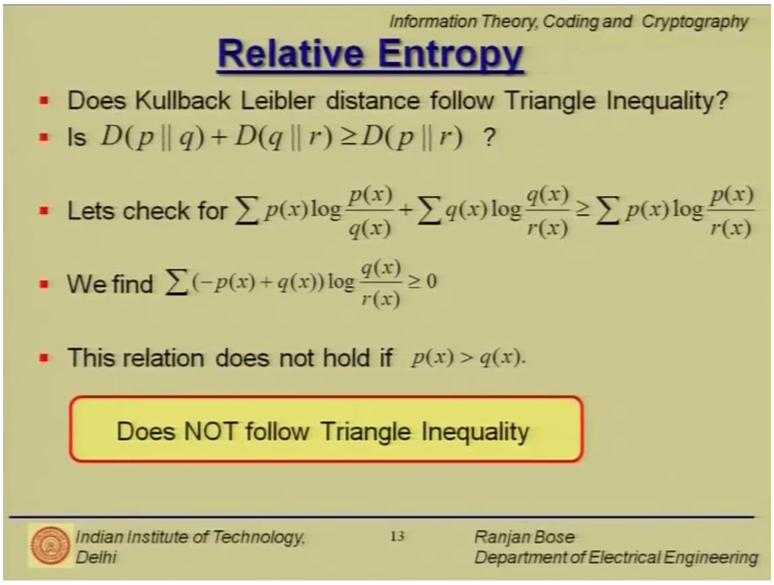


As we have seen there are clearly problems with this and our next slide will tell you, what the problems are. There are many problems. First is, if we talk about this to be a distance that is KL distance Lullback Leibler distance, then distance has 3 properties. Number 1 non-negative; so greater than or equal to 0; Number 2, symmetry property; distance between a to b is the same as b to a and triangle inequality right. The sum of the two sides of a triangle should be greater than the third side right.

So, first thing you can easily verify that the Kullback leibler distance is indeed non negative, but let us look at the other two properties. So, the question we ask ourselves is, Does the KL distance really follow the symmetry property? Just now couple of minutes back we had this discussion and we saw that there was some asymmetry. So, why do not we test it out? Question we are asking is D p parallel q equal to D q parallel p? So, we plug in the values and we check for whether this is the definition of D p parallel q. Is it really equal to D q parallel p? And if we expand it out, it does not take much effort to see that it is in really not true.

So, the first conclusion is this Kullback Leibler distance, the distance is a misnomer. It is a wrong thing to call it a distance, even though the proposals have called it a distance; it does not follow the symmetry property of a distance. What does it mean? If I say, how different is distribution p from distribution q? My answer will differ with respect to how different is a distribution q with respect to p. Nonetheless it is used in practical life.

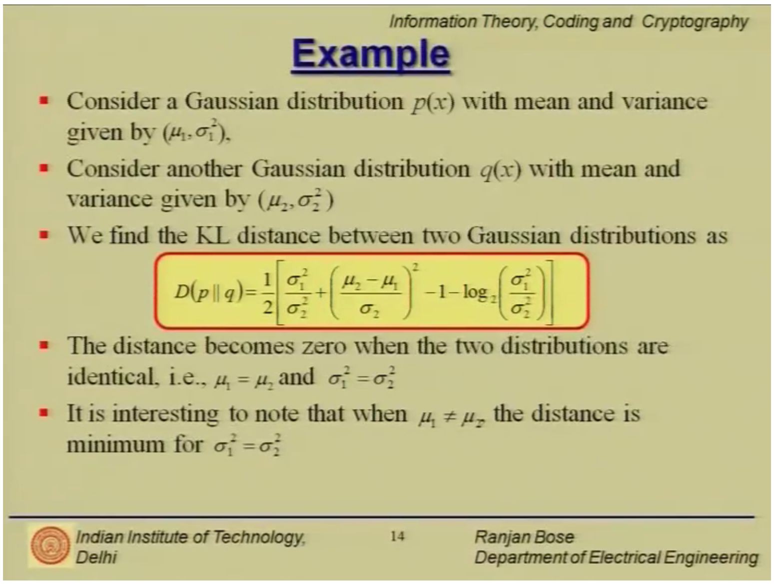
(Refer Slide Time: 21:41)



Now, we look at the second property of the distance. Does it follow the triangle inequality? Does it satisfy that condition? What does it mean? Let us take 3 distributions. So, we talk about distance between p and q, distance between q and r and distance between p and r. And the question we are asking is the distance p parallel q plus q parallel r, sum of the two sides of the triangle greater than or equal to the third side and again we plug in this values and indeed we find that for simply p x greater than q of x, we would see that this relation does not hold; it is not universally true.

So, we get that this relative entropy which is nothing, but the Kullback Leibler distance does not follow the triangle inequality. Let us look at an example

(Refer Slide Time: 22:47)



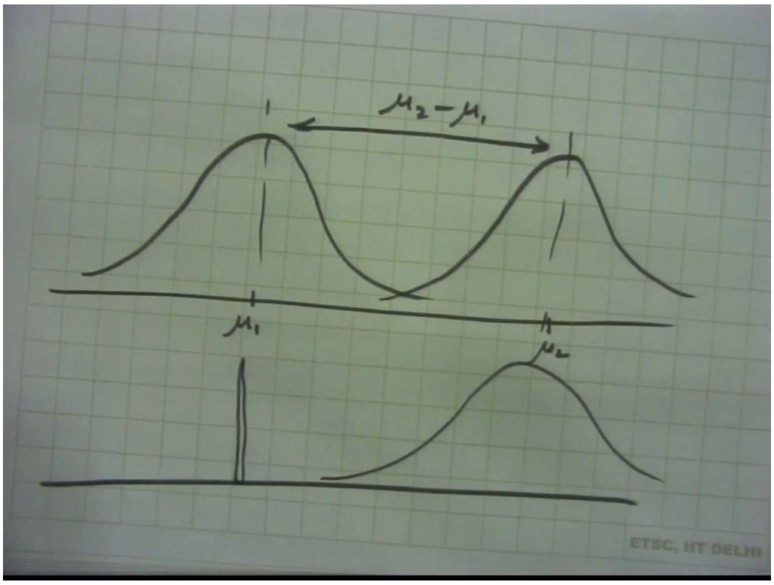
So, we are curious to find out, how similar two Gaussian random variables; call two Gaussian distributions are? So, what are my two distributions; p x and q x, then it is an interesting exercise, p x has mean mu 1 and variance sigma 1 square; q x has a mean mu 2 and a variance sigma 2 square. So, how different are they? We simply find out D p parallel q, you plug in the value and you can do a little bit of math and you get this expression.

Student: (Refer Time: 23:30) by doing you summation instead of integration using (Refer Time: 23:34) presents a plays a continuous case.

Right question being asked is, why use using summation way and integration? We have defined as a mass function, you can use integration right. As we have defined in the earlier cases. So, if you look at the two Gaussian distributions p x and q x, you get the relative entropy D p parallel q as follows. So, as expected it is a function of sigma 1 square, sigma 2 square mu 1 mu 2 and so and so forth. You will note that D p parallel q is not the same as D q parallel p. Nonetheless some interesting observations can be seen.

So, when does this distance become zero? Well obviously, when mu 1 equal to mu 2 and sigma 1 square is equal to sigma 2 square, this distribution will be 0 that is they are identical. But if you say that suppose sigma 1 square is equal to sigma 2 square, but mu 1 is not equal to mu 2 right, then the distance is minimum. What does it mean? It means that the Gaussian spread is the same.

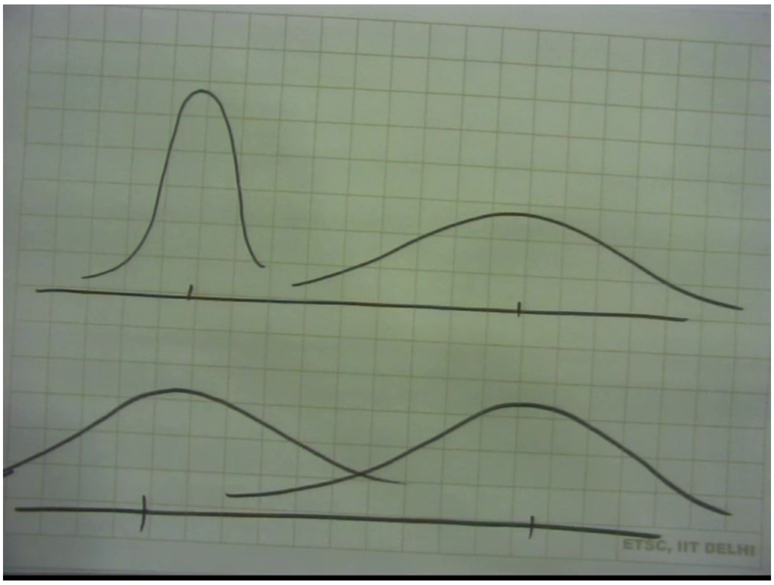
(Refer Slide Time: 25:00)



So, if you look at these two distributions. So, variance is the same, but they are apart right. In this case KL distance tells us that this is minimum and it is only a function of mu 1 and mu 2 fine. But it is minimized when the variances are the same, but if you go ahead and say that if either sigma 1 square tends to 0 or sigma 2 tends to 0, then you see; then we are in a fix because the distribution the distance becomes infinite.

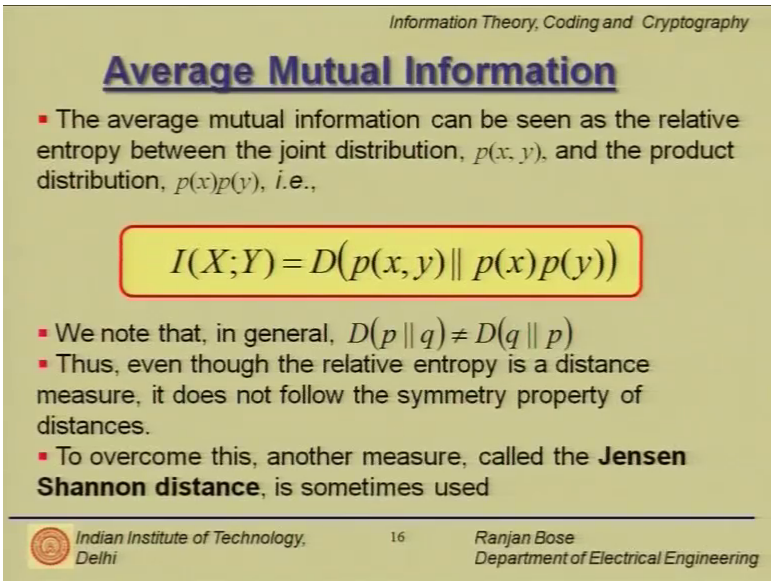
So, this is the case, when either 1 becomes in the limiting case, a delta function. So, they are very different. You can visibly see that this, these are different and somewhere in the middle if you have one distribution like this, the other one is like this; variances are different, means are different. They are clearly different.

(Refer Slide Time: 26:22)



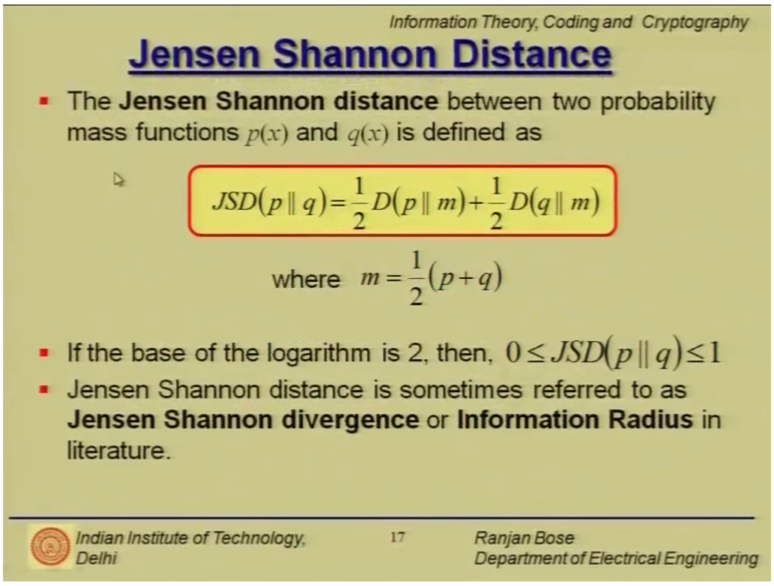
On the other side if I put variances are the same, means are different; then the distance scale distance is less. So, these two probability mass functions, it the distributions are much more similar to each other than these two distributions. Not only it is visibly correct, you can also calculate it from the expression for KL distance ok. So, there is a strong physical connotation attached to how similar or different two distributions are.

(Refer Slide Time: 27:20)



Now, we move on to the average mutual information for continuous random variable and it is a very elegant way to define it. It is I X semicolon Y is nothing, but the distance between the joint distribution and the product of the distribution. So, it is a relative entropy between the joint distribution p x comma y and the product of the distribution p x into p y. Clearly if we have p x comma y is a same as p x into p y, we get a direct obvious conclusion right. If they are independent, then the we have a one notion. But in general we know that this distance measure is not symmetric. So, Jensen Shannon came up with an alternate measure of this distance which is symmetric and let us define it also.

(Refer Slide Time: 28:30)

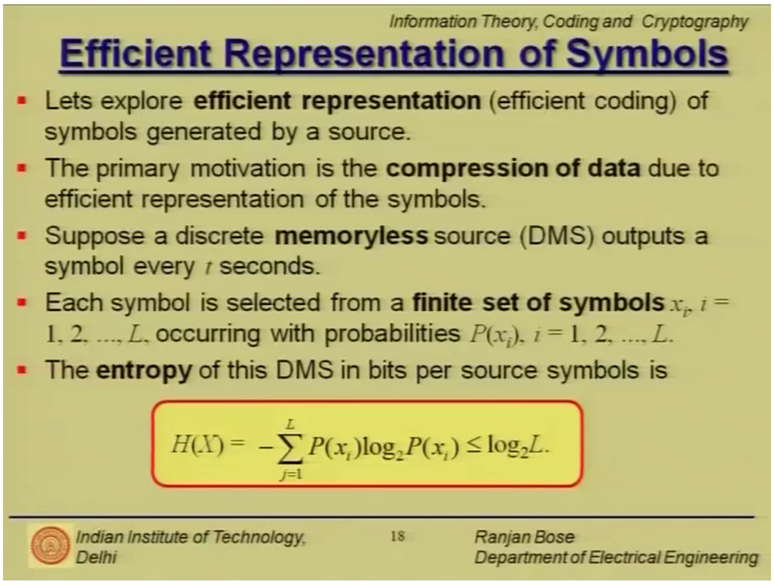


So, we are now looking at overcoming the shortcoming of the Kullback Leibler distance. We talk about the Jensen Shannon distance. Again it is between two probability mass functions p x and q x, but it is defined as JSD start standing for Jensen Shannon distance is half of D, relative entropy p parallel m plus half D q parallel m where m is an intermediate point between p and q.

So, this you can check for yourself is symmetric. So, JSD p parallel q is equal to JSD q parallel p. So, Jensen Shannon distance is also referred to as Jensen Shannon divergence or information radius and literature. So, those terms are used interchangeably. And this value is limited between 0 and 1 under the condition that the base of the log is 2.

So, why are we doing all of this? There has to be some practical utility for all of these mathematical exercises. So, far we have built in some tools to understand, what is the best way to represent symbols?

(Refer Slide Time: 30:00)

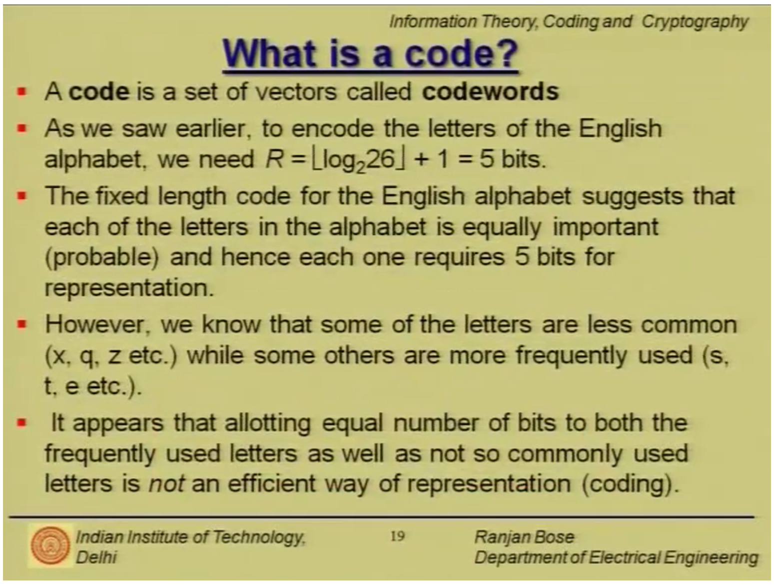


So, in the next few slides, we will explore; what are the ways to efficiently represent in other words efficiently, code symbols generated by a source? In earlier lectures, we talked about, what could be a source? It could be a man tossing a fair coin and shouting out 1 0 0 1 1 0 or he could be tossing 2 independent coins and saying 1 0 0 0 0 0 0 1 and so and so forth or he could be tossing unfair coins or he could be typing his SMS, that is a source or a monkey typing of keyboard, that is the source; all of these are sources.

They generate symbols it could be a b x y z p q or 1 0 0 1 or it could be voltages all of them are symbols for me and I need to represent them efficiently. What is the primary motivation? Data compression, efficient representation leads to compression of data. Suppose we have a discrete memory less source and its outputs assemble every t seconds. So, each symbol is selected from a finite set of symbols.

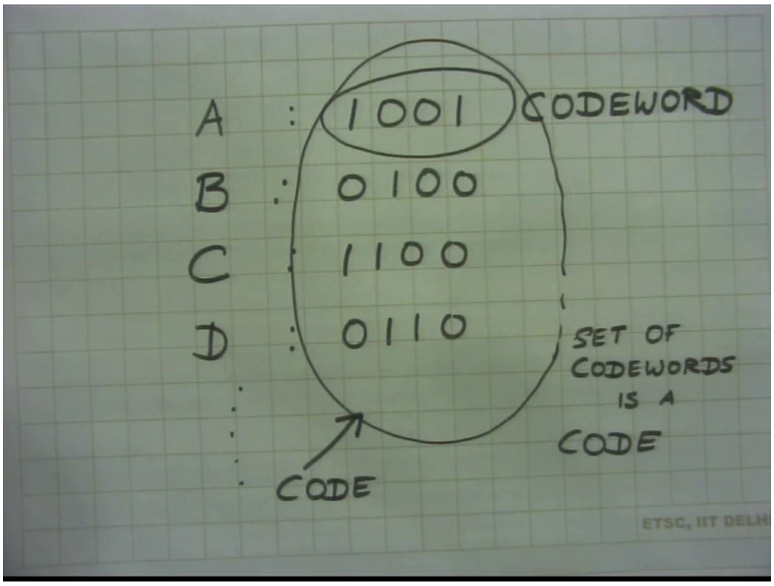
So, again we make this assumption that the set is finite, we will move to infinite sets also and we can always define the average self-information or entropy of this discrete memoryless source from the theory we have developed so far. And you can always show that this is upper bounded by log to the base 2 L. So, let us get some definitions in order.

(Refer Slide Time: 32:00)



We will represent symbols. So, suppose we want to represent the letters of the English alphabet, so A can be represented as 1 0 0 1, B can be represented as 0 1 0 0 and so and so forth. So, a vector which represents a symbol is actually called a codeword. So, if A is represented as 1 0 0 1, then the codeword for A is 1 0 0 1 ok.

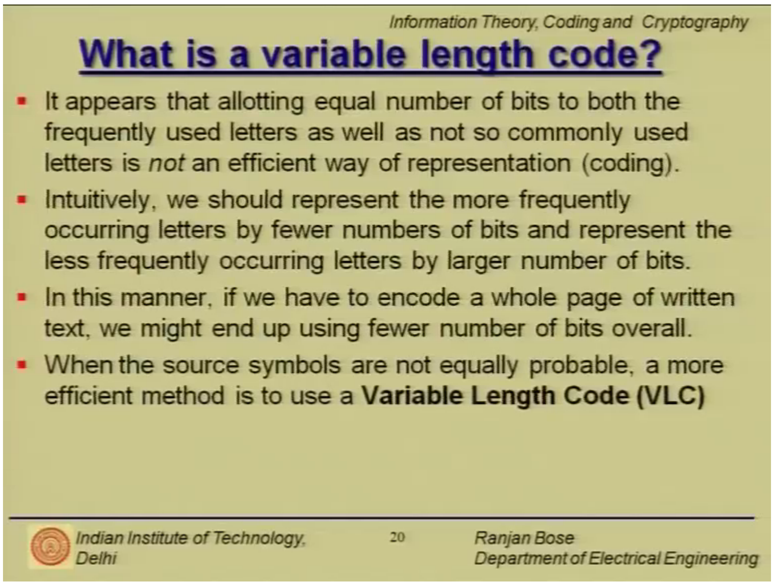
(Refer Slide Time: 32:44)



So, let us look at it in a simple manner. So, A it is represented as B and so and so forth. So, this is a codeword; however, the set the set of codewords is called a code. So, this is a code. In this example a code is a set of codewords, 1 codeword for A, 1 for B, 1 for C and 1 for D. So, code is a set of vectors called codewords. So, we have seen in an earlier example that we can encode the letters from an English alphabet and we need a certain number of bits. Here I have shown in my earlier example, a 4 bit representation, I can have a 5 bit representation clearly. If there are 26 characters in the English alphabet, then I have to have minimum 5 number of bits, otherwise I cannot have unique representation.

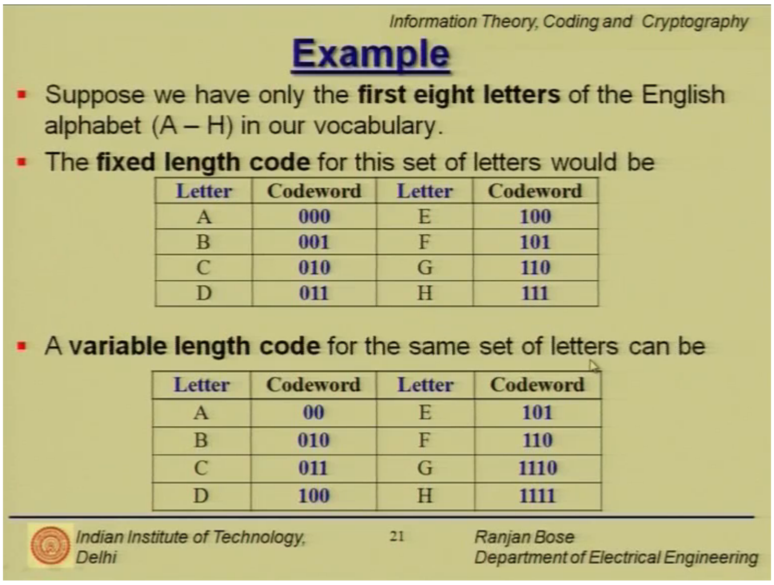
And we have also observed in our previous lectures that certain number of alphabets, certain alphabets are more frequent a, e, s, t; some are less frequent x, q, z, j and so, it does not make sense to represent all the alphabets with equal number of bits. What is the rationale behind it? Bits are expensive. To transmit a bit, I need power, I need bandwidth, I need time. All three are very precious quantities for me in my modern communication systems. If x does not appear frequently, why should I allocate certain number of bits? Or if A appears more frequently, maybe I should use fewer number of bits to represent it.

(Refer Slide Time: 35:17)



This brings us to the notion of variable length code. Remember a code is a set of codewords. So, in the example that I have shown earlier, I only had fixed length code where A, B, C, D all of them had four bits, but now maybe we should be looking at and the notion of variable length code. What is the necessity for this? When the source symbols are not equally probable, it makes sense to use fewer number of bits to represent more frequently occurring symbols and vice versa and therefore, we would use the notion of variable length codes.

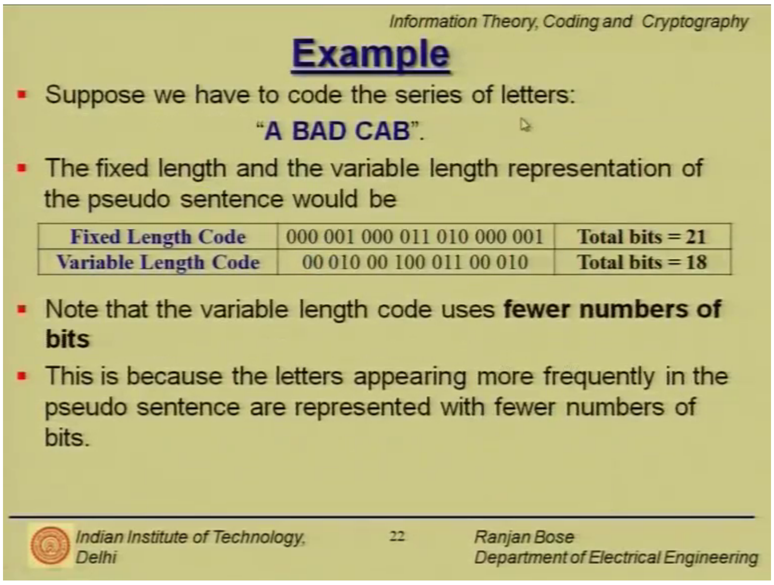
(Refer Slide Time: 36:05)



Let us look at an example. Suppose we have only the first 8 letters of the English alphabet A to H in our vocabulary. So, you go only you going to use A, B, C, D, E, F, G and H. H it is a very convenient example, log to the base 8, log to the base 2 of 8 gives me 3. So, conveniently I can have fixed length code, 3 bits per symbol right from 0 0 0 0 0 1 up to 1 1 1 and I have got the first fixed length code. It is no brainer ok.

At the same time I would say, hey how about representing them with unequal number of bits. Why do not we do a variable length code? So, I have an example. A is 0 0, B is 0 1 0, C is 0 1 1 so and so forth. I run out of certain number of bits and then I have to use 4 bits also, but I am not too bad. I am using 2 bits and 4 bits and 3 bits. So, maybe I will come out.

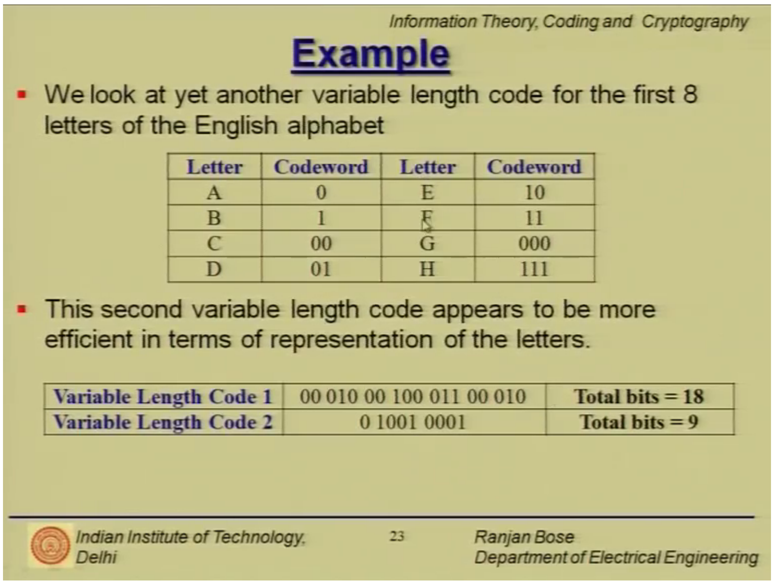
(Refer Slide Time: 37:14)



So, how about looking at a practical example? Since I have only 8 letters in my alphabet, let us make a small sentence. A BAD CAB, it only uses only 4 first 4 letters. So, if you take a bad cab and use the first code, what is a code? Is a set of codewords. So, the table 1 is a code, table two is also a code. Table 1 is a fixed length code, table 2 is a variable length code.

So, if you use the fixed length code, you have how many characters? 1, 2, 3, 4, 5, 6, 7, 8; so 7 into 3, 21 bits is what I expect from the fixed length code. But if you look at the variable length code, I have got fewer number of bits; looks like we have a winner right. We have been able to save 3 bits percentage wise that is not too bad.

(Refer Slide Time: 38:22)



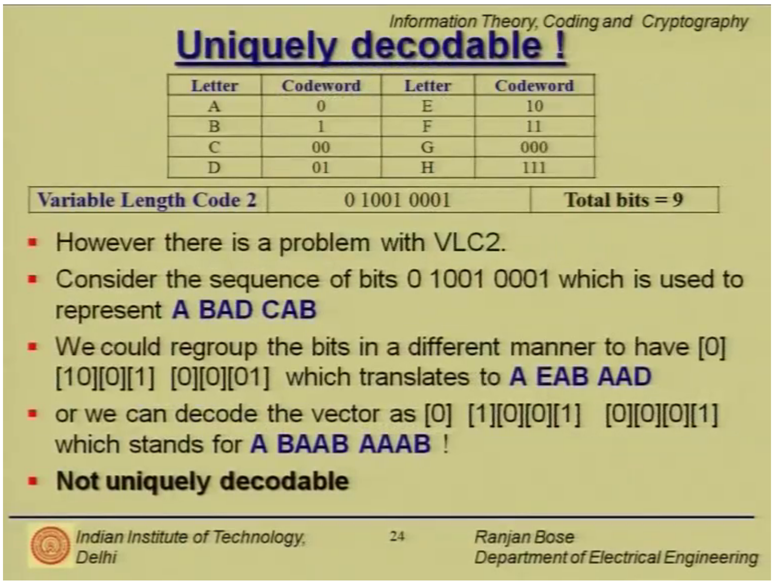
So, let us look at another variable length code for the first 8 letters of the English alphabet because this time I am really excited, I am going to squeeze out fewer number of bits will be used to represent ABC and D so and so forth.

Student: So, if it always (Refer Time: 38:40) variable length code will reduce the number of bits may be in some cases you can also increase.

So the question will asked is, does variable number of code, the variable length of codewords always reduce the representation? So, answer depends on a, how efficient is our code and b, what is the frequency with which the letters are appearing. So, we will give an example where a variable length code can actually lead to expansion and not a compression. So, that can also happen, but we talk about on an average. Yes, on an average a variable length code.

So, the most 3, most important 3 words are on an average. What are you saying on an average? Yes on an average a variable length code, if designed properly will be able to compress, but once in a while for a very special set of input characters, it can lead to expansion. But hey we carry out our communication over millions of bits and it averages out. So, we turn out a winner.

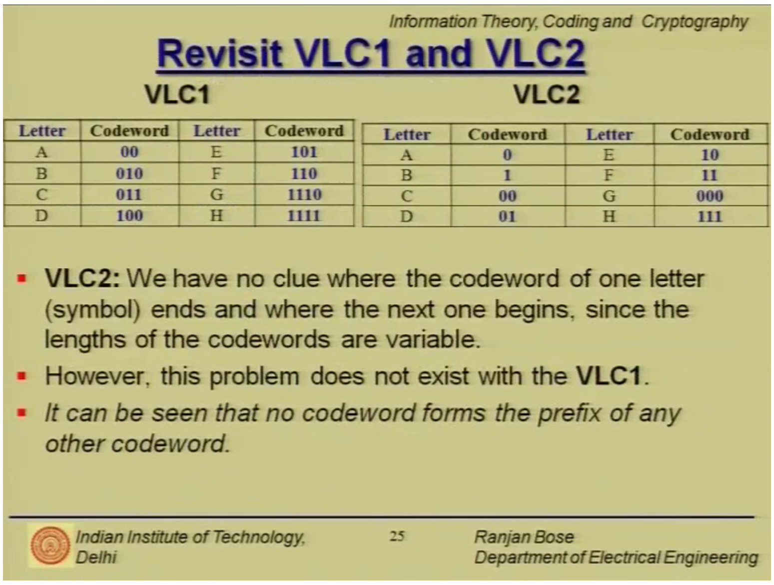
(Refer Slide Time: 40:00)



So, if you complete this example, if this much more constricted code A only 1 bit, B only 1 bit, C 2 bits and so and so forth and only when I am first I am run out of 1 and 2 bits, I go to 3 and 3 bits. And then I again encode a bad cab, I just have a 9 bits, but the problem is the decoding part. You cannot decode it.

At least not uniquely because if you look at it a bad cab is truly 0 1 0 0 1 0 0 0 1, but it can be broken up into different sets and it becomes A BAD AAD or A BAAB AAAB. So, clearly it is not uniquely decodable. Since I do not know a priori, how long are the codewords; since they are variable length? I do not know at the decoding end. What was actually sent? I am in trouble unless I have a smart way to overcome this deficiency.

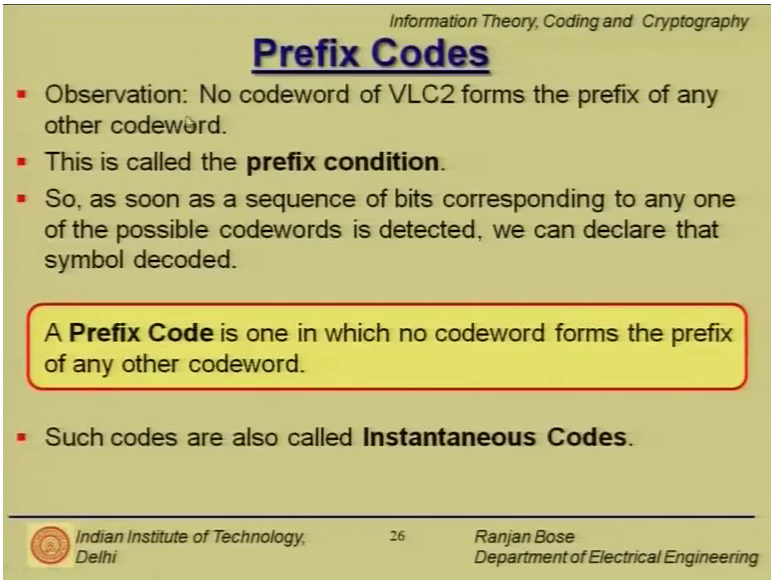
(Refer Slide Time: 41:13)



So, let us revisit variable length code 1 and variable length code 2. If you remember variable length code 1, did give us a reduction, but not by much and variable length code 2 give us a major reduction, but it could not be uniquely decoded. So, you look at variable length code 1, can it be uniquely decoded? Well the answer is yes. The answer is yes because if we make a simple observation that no codeword is a prefix of any other codeword.

So, variable length code 1 has a very very unique characteristic. What is it? No codeword is a prefix that is no other codeword starts with any other codeword. No codeword is a prefix of any other codeword and hence decoding is absolutely unique and instantaneous. The moment you find a valid codeword has come, you declare the result because there is no point in looking further because no codeword is a prefix of any other codeword.

(Refer Slide Time: 42:37)

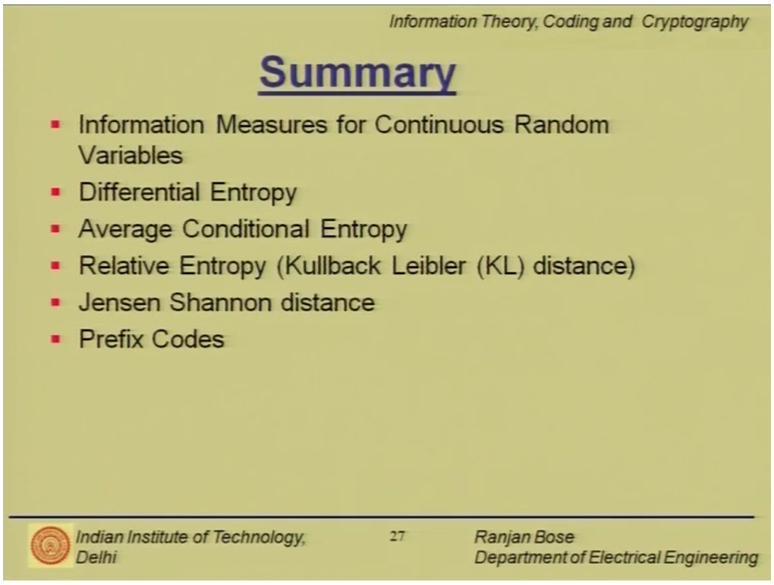


This interesting property is called the prefix condition and the decoding strategy is very simple as soon as a sequence of bits corresponding to any one valid codewords is detected, we declared the resemble being decoded.

So, we now formally define, what is a prefix code? A prefix code is one, in which no codeword forms the prefix of any other codeword and since I can instantaneously declare the results as we go along. These are also called instantaneous codes. So, let us summarize what we have learned today.

We started with information measure for continuous random variables; we made a distinction between discrete random variables, continuous random variables. We could extend the definition, but not the physical interpretation because the average self-information of contained in a continuous random variable is actually infinite.

(Refer Slide Time: 43:48)



We then talked about differential entropy and we observe that even though, you cannot really talk about in real terms; the information content of a continuous random variable. You can talk in terms of a differential mode. So, X is a continuous random variable, Y is a continuous random variable. So, h X minus h Y may not be infinite. Even though h of X and h of Y, each of them are infinite and hence this word name differential. So, it only makes sense any meaning, when it is taken as a difference of 2 h X and h Y ok; hence the name differential.

We talked about the average conditional entropy for continuous random variable, then we have raised the very interesting question. How do you say two probability distributions are similar or different? What is the similarity measure? And we talked about relative entropy is also called the Kullback Leibler distance, we also observed that it is a distance measure, but it is a pseudonym, it is a misnomer only the non-negativity is satisfied. It does not follow the triangle inequality nor does it follow the symmetry property of a distance measure.

So, to overcome that we talked about the Jensen Shannon distance which is symmetric and finally, we introduced the notion of prefix codes; that is the first step towards efficient representation of symbols and ultimately we have look at ways to compress data, speech, images what have you right. So, that is one of the fundamental contributions of source coding. It let us you calculate the theoretical limits to which I can compress my data and no further. That is where we will go in our next module.

Thank you.