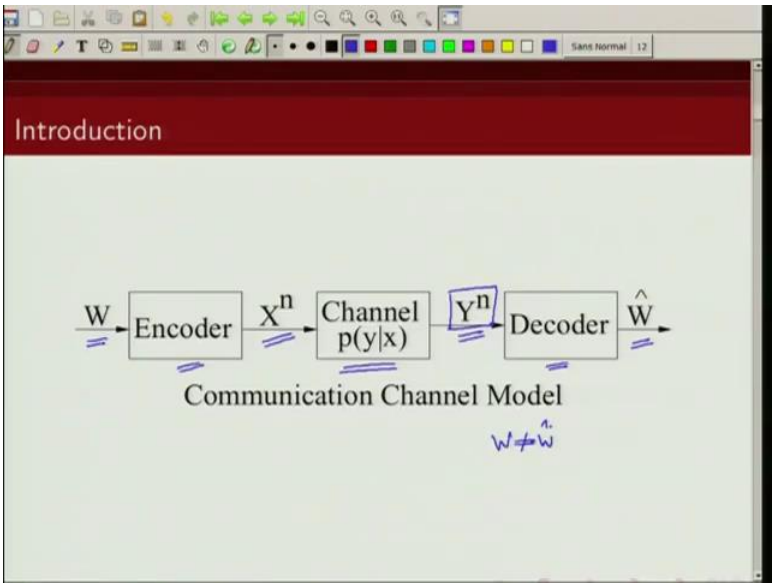
An Introduction to Information Theory
Prof. Adrish Banerjee
Department of Electronics and Communication Engineering
Indian Institute of Technology, Kanpur

Lecture - 10B
Noisy Chanel Coding Theorem

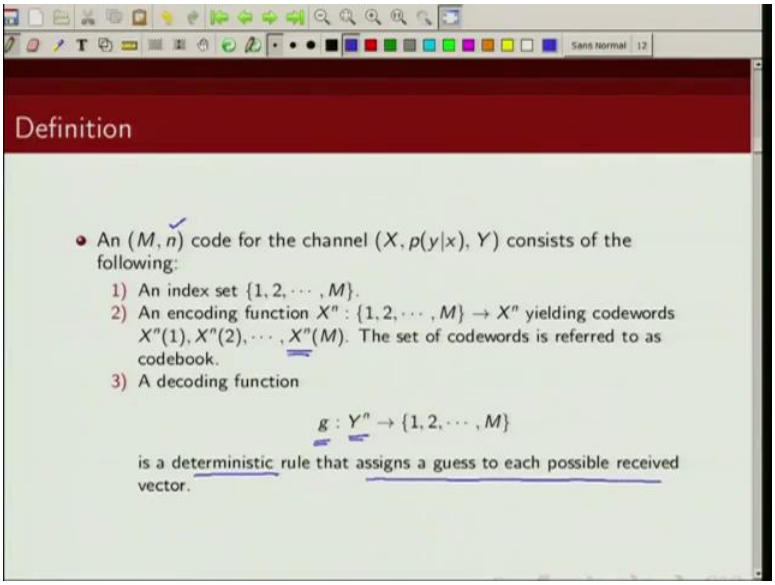
Welcome to the course on an Introduction to Information Theory. Now in this lecture we are going to talk about Shannon's noisy channel coding theorem. We are going to proof the achievability part of the result and we are also going to prove the converse of this theorem. In the last lecture we have already build up the background to prove our noisy channel coding theorem.

(Refer Slide Time: 00:46)



So, we start our lecture with some basic definition and model of the communication channel that we will consider. So, w is have a message that you want to send is encoded using this encoder and x of n is the code word, that is transmitted over a discrete memory less channel it is transition probability channel transit probabilities are given by probability of y given x, yn are the received noisy code words and the job of the decoder is to estimate what was the message signal that was transmitted given received sequence yn.

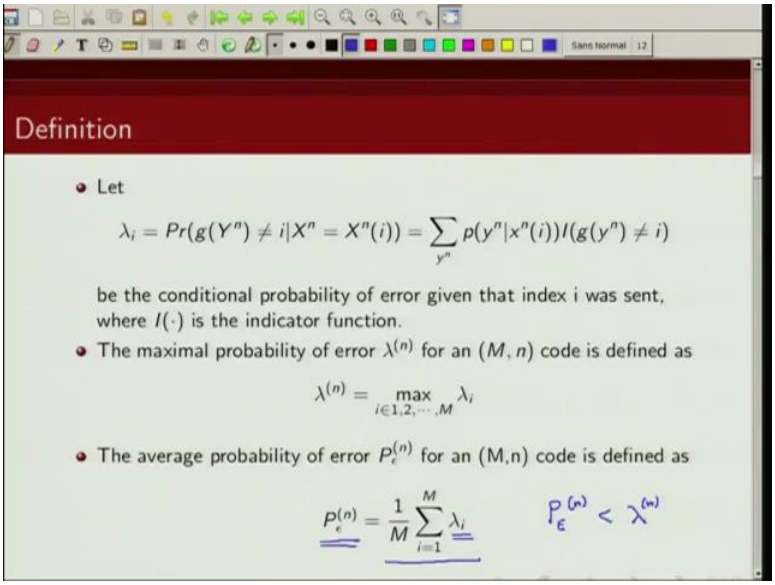
(Refer Slide Time: 01:49)



Now, an error happens if w is not same as w hat otherwise there is no error. So, this is the communication channel model that we have. That we are considering was defined few terms before i go in to the noisy channel coding theorem. So, we define a code by these parameters m and n i will explain these terms in a little while and discrete memory less channel is described by this input x output y and channel transient probability which are given by probability of y given x. So, this index set will be you basically used to denote will be correspond to each of these messages, and what we are going to do encoder is a mapping of this in index set to a n bit sequence and this we are denoting by x of n.

So, we will have m code words this is xn1 xn2 denoted by xn3, xnm. So, these are our set of code words, and we will refer to a set of code words as code book. Now as you saw once you send these x ns at the channel output at the receiver input you are receiver you will received y of n. So, once you receive y of n then you essentially need to find out which of these are messages you have transmitted. So, decoding is essentially to be operation. So, it is a deterministic rule that assigns a guess to each possible received vector n is the length of the codeword and m are number of code words that you have.

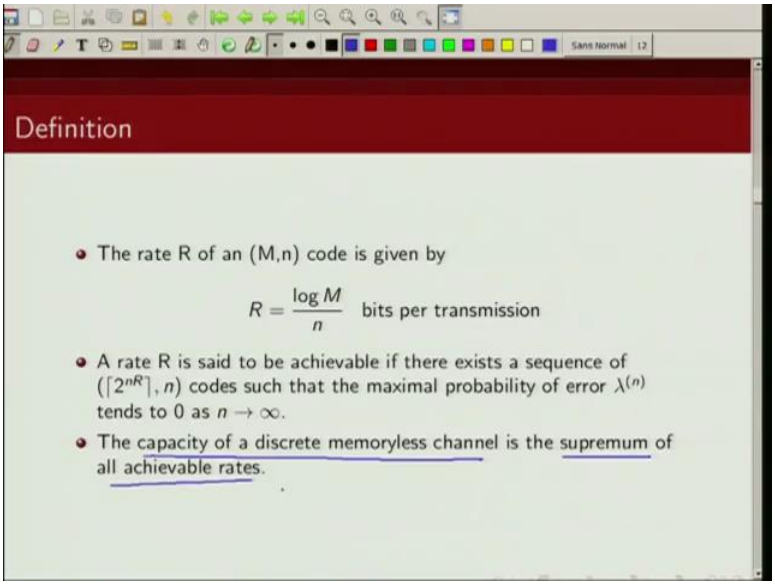
(Refer Slide Time: 03:41)



Lambda i is nothing, but conditional probability of error given that index i was sent. So, i sent index i what is a probability of error. Now when will errors happen if the decoded. So, decoder function is denoted by g. So, if g of y of n is not same as i and when you transmit ith index. So, when x of n is x x of n i when you transmit the ith index, that your decode are does not decoded as ith index then the error happens and that is given by this you know this is the indicator function indicating, whether error has happened or not and this probability yn given x x of n i. So, this is. So, lambda i is a conditional probability of error given that you transmitted the ith index. Now you define maximal probability of error as maximum of lambda i over all possible indexes.

So, index i can go for 1 to m. So, maxima of lambda i that is a maximal probability of error similarly we can define average probability of error. So, lambda i is the error, when ith index is sent. So, when you serve overall lambda is and divide by m you get average probability of error is not very difficult to show that this average probability of error will be less than this maximal probability of error. Now we are going to show you later in this lecture that as long as a transmission rate is below channel capacity our average probability of error is about, is very small. It goes is to 0 as block length codeword length increases and we will show if the average probability of error goes to 0 then the maximal probability of error also it goes to 0 as then becomes large.

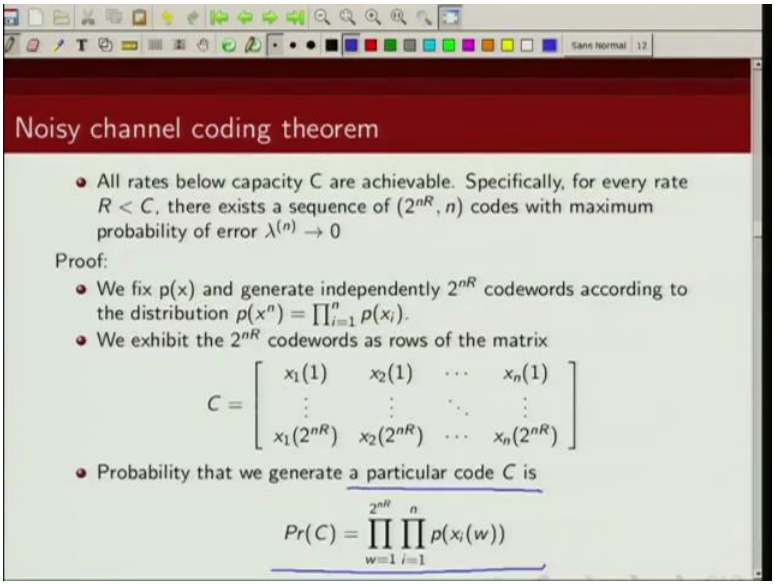
(Refer Slide Time: 06:09)



So, as I said that total m indexes that we are sending. So, that number of information bit's or log of m and the codeword length is n. So, the rate of the code is denoted by log of m divided by n.

Now, we see a rate is achievable if there exists a sequence of codes that parameter m given by seal of 2 raise to power n R n of length n such that the maximal probability of error goes to 0 as n goes to infinity. So, we say a rate is achievable if for that particular code with parameter 2 raise to n R n the maximal probability of error goes to 0 as the codeword length increases and we defined the capacity of a discrete memory less channel as the supremum of all achievable rates. So, so supremum is the smallest it is the upper bound of a set.

(Refer Slide Time: 07:36)



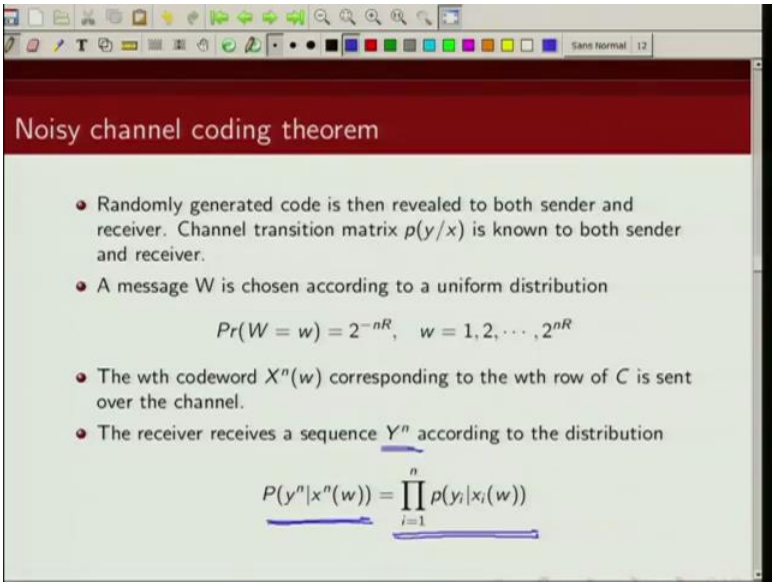
So, capacity is basically supremum of all achievable rates, now for a discrete memory less channel Shannon's noisy channel coding theorem says that all rates below channel capacity are achievable. That means, if we transmit at rate below capacity then our maximal probability of error will go to 0 as the codeword length increases to infinity. So, so specifically for every rate which is less than capacity there exists a code of these parameters. So, if that maximal error probability goes to 0.

Now, we are going to prove that cumulative part of this theorem first. First we are going to show that average probability of error goes to 0 as n become large and, then we will show that if the average probability. Whenever goes to 0 then the maximal probability of error is also bounded and it is it goes to 0. So, first we do is we fix our input distribution and generate 2nr code words according to this distribution i had a distribution and we considered a rows matrix, where each row is basically randomly generated code words. So, this x x1 and x2 and x n is codeword similarly these. So, we generate 2 raise to power 2 n R code words and they are start up in a matrix like this. This is randomly generated.

So, probability that we generate a particular code C is given by this probability. So, probability of x i w, where is w can go for 1 to 2 raise to power n R and i goes from 1 to

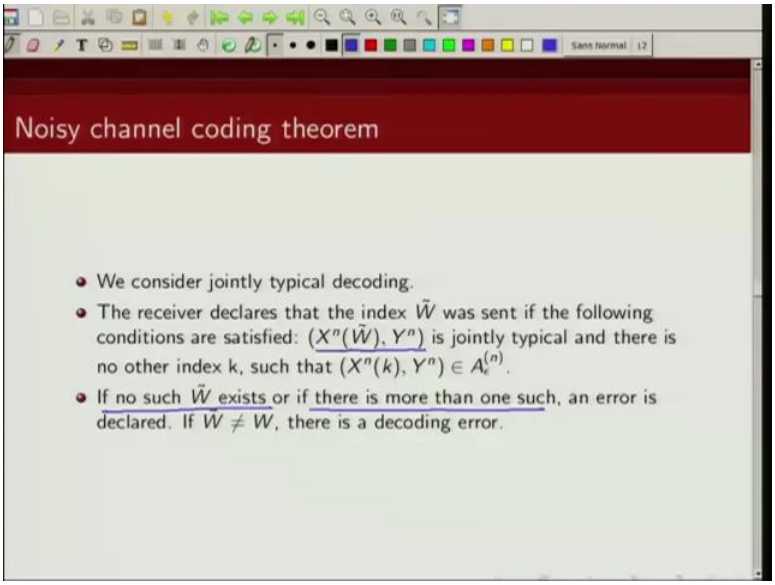
n.

(Refer Slide Time: 09:40)



Now, once we randomly generate a code we reveal this code to both the sender and the receiver now we assume the channel transition matrix probabilities are both known to the sender and receiver. And we chose a message w according to a uniform distribution. So, probability that we chose a particular index is given by1 by 2 raise to power 2nr. So, that is this probability and the wth codeword corresponds to the wth row of this code matrix that we talked about. So, when we select this index w we are actually sending this wth row of this codeword matrix, that we talked about that we are randomly generating now what the receiver receives is y of n and probability y of n given a particular x n of w was sent this probability for a discrete memory less channel with the feedback is given by this expression.

(Refer Slide Time: 11:02)



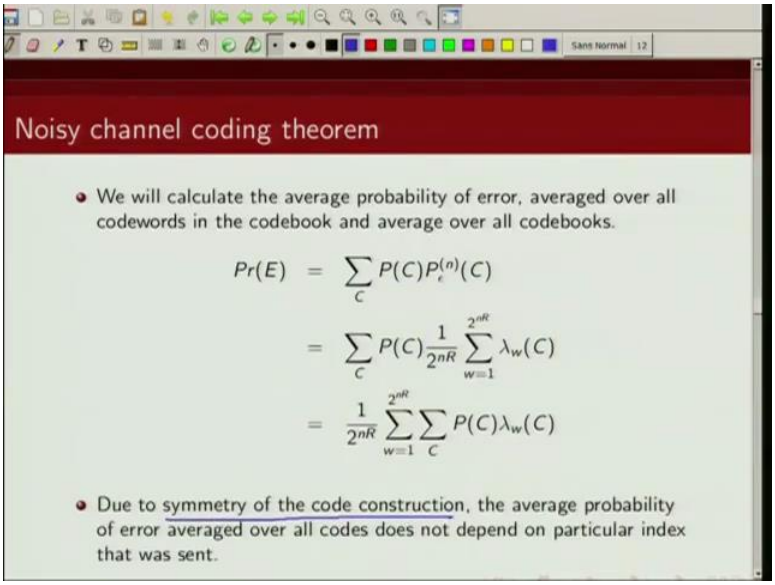
Now, once we receive yn at the receiver we are going to do, what we call jointly typical decoding. So, we are going to do a typical set decoding now this is a suboptimal decoding technique, but asymptotically this is basically optimal. So, how does a typical set decoding works it works as follows. So, the receiver declares that an index w tilde was sent if the following conditions are met and what are those following conditions if x n w tilde and yn which is a received sequence. If this is jointly typical and there is no other index k such that x n k and receives sequence y of n are jointly typical. So, we you say that we successfully decode and index if x n w and that index and y of n they are jointly to become and there is no other index for which x n k and yn are jointly typical and that is basically your typical set decoding.

So, when will an error happen an error will happen when, you send w index and you decide in favor of some other index which is not w. So, an error can happen in a typical set decoding if there is no w tilde; that means, there does not exist any index for which this pair of that x n w tilde y it is jointly typical that is 1 condition or the second condition under which an error can happen is as follows. If there is more than what 1 such index for which x n k and yn is jointly typical, so, there are 2 possible cases of error 1 if there does not exist any index w tilde such that x n w tilde and yn are jointly typical that is 1 condition second condition is there exist more than 1 index for which x n k and

y of n are jointly typical then also you will make an error decoding error.

So, let us try to calculate what is the average probability of error and we are doing calculating this average probability of error average over all possible code words and averaged over all possible code books. So, this a probability of error for particular code book and is a probability of this codeword C we average it over all possible code books.

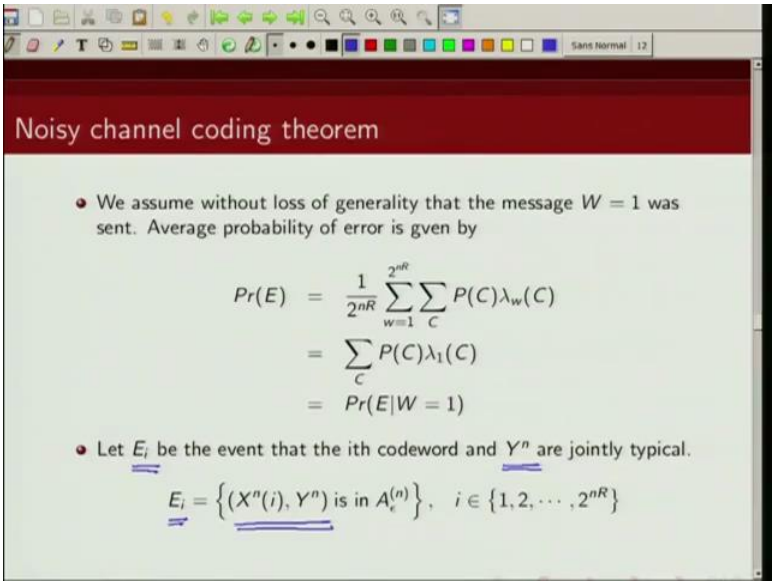
(Refer Slide Time: 14:02)



So, this probability of error from definition this is given by this as you recall 1 byn summation w 1 to n lambda w C right. So, this can be written like this now please note that this term does not depend on index w why, because there we are constructing our code due to symmetry of the code construction the average probability of error average over all possible codes is not a function of a particular index. Now what is that mean it means that then, we can calculate our probability of error given a particular index has been send since it does not depend on what index probability of error does not depend on what index have been sent.

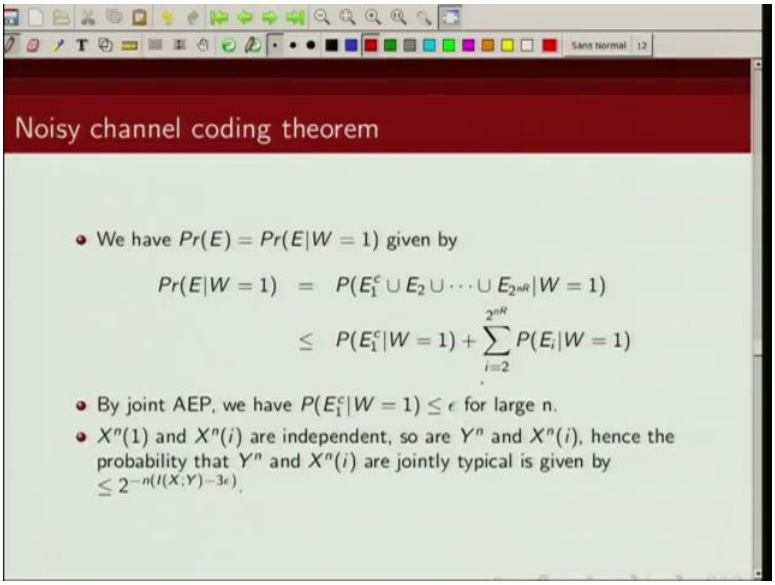
So, we can calculate a probability of error assuming a particular index i was sent and therefore, we are going to do.

(Refer Slide Time: 15:32)



So, without loss of generality we will assume that, let us say the first index was sent and we are going to calculate probability of error given the first index was sent. So, the probability of error can be given by probability of error given the first index was sent and that is probability of error given index1 was sent. So, let us compute this probability now let us denote e i to be the event that ith codeword and received sequence y of n are jointly typical. So, e i is the event that x n i and yn belongs to jointly typical set now as I said probability of error does not depend on what index has been transmitted. So, we do analysis assuming; let us say index1 was sent. So, what is the probability of error, when index 1 was sent?

(Refer Slide Time: 16:29)



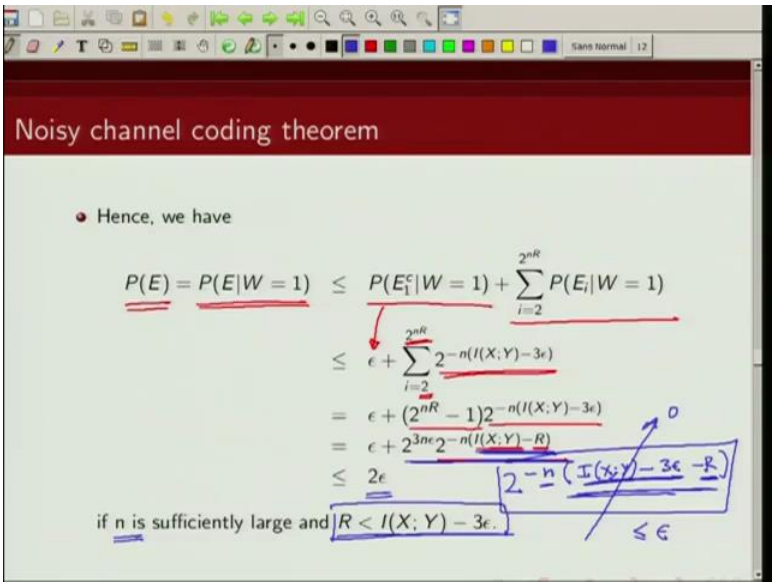
So, this is given by. So, as I said this is given by probability that what is E1c? E1c corresponds now if index 1 was sent the decoder will not make an error if xn1 and y of n are jointly to become and that corresponds to event e one. So, if e 1 compliment happens then that is an error because if index1 has been sent the decoder will not make a mistake if event e 1 occurs. So, an error will happen if the complement of the e 1 e event even has happened or error can happen if any of the other index, which was not sent is jointly typical with the received sequence y of n. So, for example, if yn is with xn2 or xn3 then there is an error because index1 was sent. So, you will. So, I can write upper bound probability by i can write it as probability of e 1 complement union of e 2 event union of e 3 event union of up to e of 2nr given, that first index was sent and usually union bound. I can upper bound it then as probability of e 1 C complement given index1 was transmitted plus probability of e i given w index1 was transmitted and i go from 2 to 2 n r.

So, this corresponds to the event that yn is not jointly typical with it. If you go back and see this corresponds to event e 1 corresponds to the e event that x and 1 and yn are jointly typical. So, this corresponds should even event that xn1 and yn are not jointly typical and this corresponds to the event that any other index. Let us say xn2 xn3 x n 4 they are jointly typical with y of n now. So, then the error probability is upper bounded

by this probability and this probability, now from the property of joint a e p we know that if n is very large then x of n 1 and y of n are likely to be jointly typical. We have shown property of jointly typical sequence that probability that x of n 1 and y of n belongs to a typical jointly typical set that probability goes to 1 as n goes to infinity. So, then this probability that event e 1 complement happens given that w is 1 this probability is going to be very, very small less than epsilon.

Now, let us look at this now we have also shown that if, now if xn1 and x n i where i is different from 1 they are independent. So, yn and x n i are also independent with probability that yn and x n i are jointly typical this probability is upper bounded by2 raise to power minus n times mutual information minus 3 epsilon this also follows from the property of joint typical sequence this we have proved in the previous lecture. So, when x n i n and y of n are independent probability that they are jointly typical this upper bounded by this.

(Refer Slide Time: 21:37)



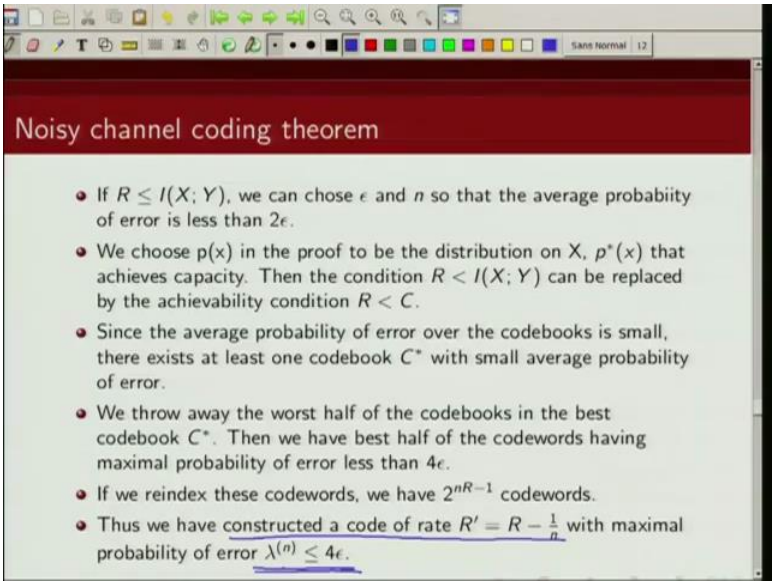
So, then we can compute the probability of error as we said it is independent of what index we have sent that is because the symmetry of code construction this is upper bounded we have shown by probability of the complement of event e 1 happening, given index 1 was sent plus this probability now this from the property of joint 80 is a very

small quantity you will calling it epsilon this we know is upper bounded by this probability.

So, when we sum it above from i equal to 2, 2 raise power n R what we get is is equal to epsilon plus 2 raise to power n R minus 1 times this quantity now we just collect terms. We collect terms containing epsilon. So, this will become epsilon plus 2 raise to power 3 n epsilon times 2 raise to power minus n mutual information minus r. Now as real as our transmission rate are is less than this quantity mutual information this particular term is going to be greater than 0 and for large n. So, this is this is this term is greater than equal to 0 if a large n 2 raise to power minus n this term will go towards 0.

So, if n is sufficiently large and our weight is less than see this we can combine as 2 raise to power minus n minus 3 epsilon minus R this is what you will get. So, as long as R is less than this quantity epsilon is very small this term will be positive for large n this whole term will go towards 0. So, some epsilon we calling it, so this will be less than some small quantity epsilon. So, what we have shown here is probability of error that is probability of decoding making a decoding error is less than 2 times epsilon provided our transmission rate is below this mutual information in x and y minus 3 epsilon and provided this length n is very large if n goes to infinity.

(Refer Slide Time: 24:34)



Now, we can tighten this a bit. So, if you take are to be less equal to mutual information and we can choose our epsilon and n such that probability of error is less than 2 times epsilon. Now we can now choose our p of x remember we are generating our code words using this distribution p of x, now we can choose our p of x such that we choose our distribution that we maximize mutual information. So, we can choose a distribution i am calling it p star x that will achieve it is capacity then this condition R is less than mutual information can be replaced by the condition that R is less than capacity. So, what i have shown you. So, far is average probability of error over all possible code words is very small it is basically less than equal to 2 times epsilon now, if the average error probability is small; that means, there exist at least 1 codeword whose probability of error maybe less than the average probability of error right and remember in the beginning we said that if average probability of error goes to 0 we can also show that maximal error probability will also go to 0.

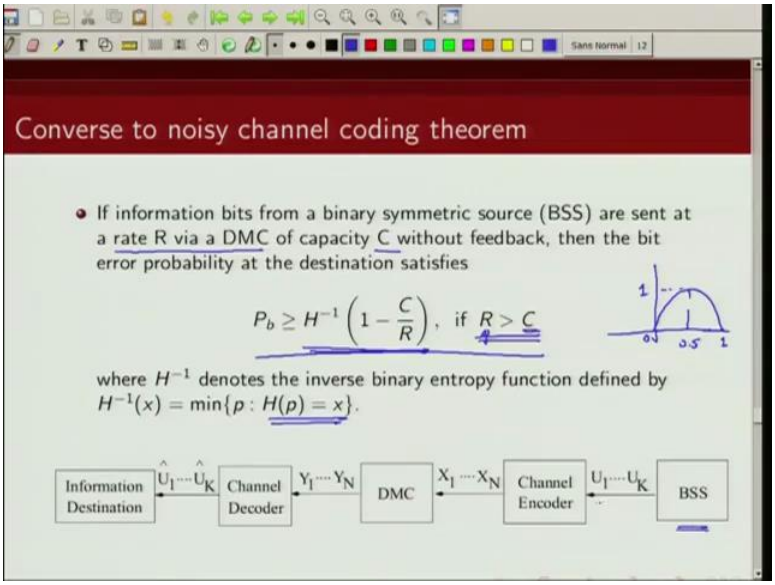
So, that is what we are going to show now or what we have shown. So, far is the average probability of error provided the transmission rate R is less than capacity is less than equal to 2 times epsilon, now what we do next is as follows we throw away half of the code words and what are these half of the code words we threw away worst half code words. So, we throw the code words which causes larger error. So, we throw half of them now we are left with other half which consists of good code words and the good i mean the once which will cause less probability of error now it can be shown that for these half code words the maximal error probability is less than 4 epsilon. So, for these best half code words the maximal error probability is less than 4 epsilon the reason being if it is more than 4 epsilon then, the average probability of error overall bad and good code words cannot be less than 2 times epsilon.

So, by throwing away half of the worst code words we have shown with the half the number of code words left we have shown that the maximal probability of error is now bounded by 4 times epsilon now since we have thrown away half of the code words. So, we can reindex these code words and now we have 2 raise to power n R divided by2 that is 2 raise to power n R minus 1 code words; so now, we have constructed a code with rate i am calling the rate R dash which is R minus 1 by n whose maximal probability of error is less than 4 times epsilon. So, if n is very large. So, this hardly any rate loss and

you can see the maximal probability of error goes to 0 as that side is large because maximal error probability is bounded by upper bounded by 4 times epsilon.

So, this proves that when our rate R is less than capacity then, we can reliably communicate because we have shown that the maximal probability of error is bounded by 4 epsilon next we are going to show the converse of this theorem now what is the converse of this theorem. So, we are going to show that if our transmission rate is above channel capacity; that means our R is greater than C then probability of error is bounded away from 0 so.

(Refer Slide Time: 29:27)

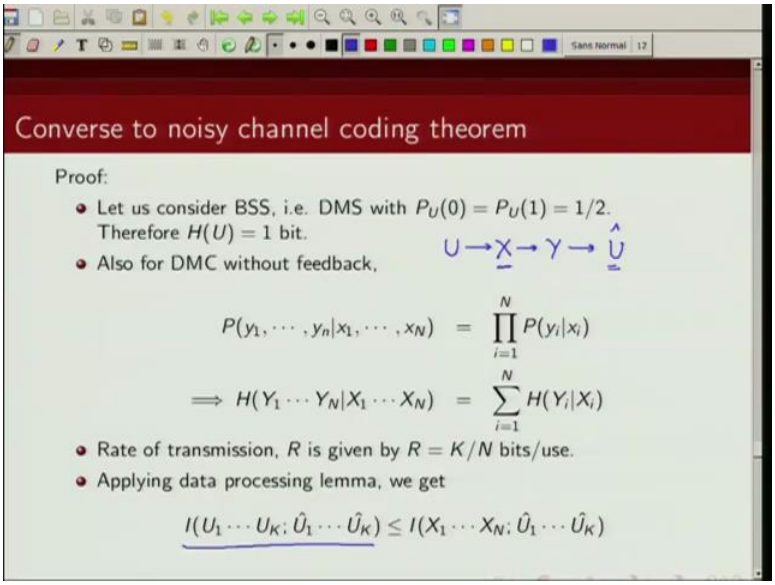


So, we will show that if transmission rate R is more than channel capacity then, probability of error is non 0. So, if information bit's. So, we consider a binary symmetric source if information bit's from binary symmetric source are sent at a rate R via discrete memory less channel whose capacity is C then probability of error is never bounded by h inverse 1 minus C by R where the transmission rate is our capacity.

So, if the transmission rate is above channel capacity then probability of error is lower bounded by quantity which is non0 and this h of p is nothing but our binary entropy function if you recall binary entropy function looks like this. So, here 2 1 at 0.5 this is

one. So, to proof this result let us look at the block diagram. So, we have a binary symmetric source output of that are u is u1, u2, u3, uuk and these bit's are sent to a channel encoder that generates these code words x1, x2, x 3, xn. So, the rate of the code is k byn and these code words are sent over a discrete memory less channel what we receives is y i. So, you see y1, y2, y3, yn. Now these are sent to a channel decoder which will try to estimate what were the information bit's that we have sent. So, the channel decoder will try to estimate u1, u2, u3. So, these estimates I am denoting by u1 hat u2 hat uk hat and this is my. So, let us now show that if the transmission rate is above channel capacity then the probability of error is lower bounded by a non 0 quantity.

(Refer Slide Time: 31:44)

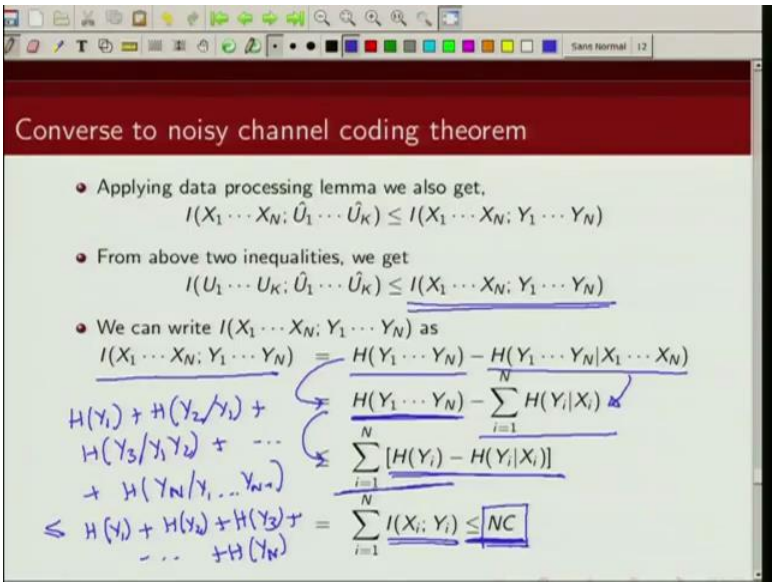


So, since we are considering a binary symmetric source. So, probability of 0 and probability of 1 is half it is a symmetric source and it is a binary source. So, it is sum is 0s and ones if probability has. So, we can write uncertainty u as 1 bit. Now for a discrete memory less channel without feedback probability of y1, y2, y3, yn given x1, x2, x3, xn can be given by probability of y i given x i and product taken from i go from 1 to n. So, we can write the uncertainty in y1, y2, y3, yn given x1, x2, x3, xn by summation of uncertainty of y i given x i and remember our transmission rate is k by n bit's per use.

So, we are going to first apply data processing lemma, what does data processing lemma

says that further processing of data does not increase information. So, what you had was we had u x y and u hat follow mark of chain. So, mutual information between u is and u hat is this is going to be less than mutual information between x is and u i hats. So, that is what i am writing here next again we will apply data processing lemma. Let we write down the mark of chain. So, u x y and u hat follows a mark of chain.

(Refer Slide Time: 33:24)



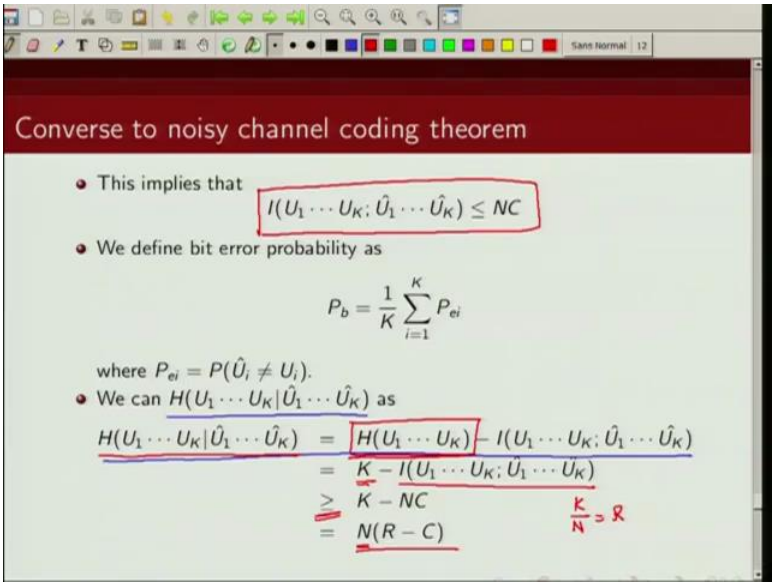
So, we have shown the mutual information between u is and u i hat is less than mutual information between x i and u i hat. Now we can say that mutual information between x i and u i hat is less than mutual information between x i and y i again that follows from data processing lemma. So, if we combine these 2 result let us call it a and call it b if i combine a and b what i get is this condition that mutual information between u1 u2 uk and u1 hat, u2 hat u k hat this is less than mutual information between x1, x2, x n and y1 y2, yn next.

Let us try to simplify this term mutual information between x1, x2, x3, xn and y1, y2, y3, yn now from the definition of mutual information i can write this as joint entropy of y1 y2 y3 yn minus conditional entropy of y1, y2, yn given x1, x2, x 3, x of n now since this is a discrete memory less channel without feedback. This term can be written as summation of uncertainty in y i given x i. So, i can write this term as joint entropy minus

the summation from i goes from 1 to n h of y i given x i. Now this joint entropy can be written using symbol as h of y1 plus h of y2 given y1 plus h of y3 given y1 y2 plus h of yn given y1 y2 yn minus 1. Now this can be further written as this is upper bounded by h of y1 plus h of y2 plus h of y3 plus h of yn.

Now, here I used the fact that conditioning cannot increase entropy. So, if I do that then I can write that this joint entropy is upper bounded by summation of entropy of h y, i where i goes from 1 to n. So, then from here I can write this expression as summation i goes from 1 to n h of y i minus h of y i given x of i and what is this term this term is mutual information between x i and y i. So, then I can write this as mutual information between x i and y i and mutual information between x1, y1, x2, y2 that is less than channel capacity.

(Refer Slide Time: 37:46)

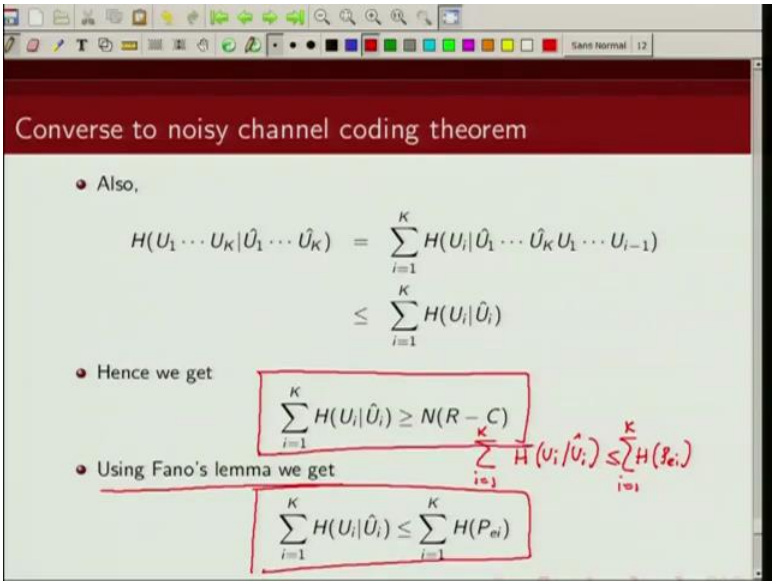


So, I can write then mutual information between x1, x2, x3, xn and y1, y2, y3, yn this is less than equal to n times C next. Now we know that from data processing lemma we know that this quantity is less than this quantity and this is less than n times c. So, mutual information between u1, u2, uk and u1 hat, u2 hat, uk hat this will be less than equal to n times c. So, combining these results we get this next.

That is defining probability of bit error. So, basically it is this is some all errors divided by the block size that we give me probability of bit error and probability of error when does an error happen when, my transmitted date is not same as the decoded date. Now let us try to write uncertainty in u1, u2, u3, uk given u1 hat, u2 hat, uk hat and these from the definition of mutual information can be written in this particular form what is this quantity the u is are the output of a binary symmetric source and this is independently identically distributed. So, u h of u1, u2, u3, uk is a h of u1 plus h of u2 plus h of u k and h of ui was 1 bit because is a binary symmetric source. So, this particular term will be equal to k k minus this now we have proofed earlier that this mutual information is less than equal to n time's c. So, if we subtract the larger quantity this will become greater than equal to.

So, then this is greater than equal to k minus n times C and what is k by n is r. So, k is n times r. So, when we write k is n time R and we take n out what we get is uncertainty in u1, u2, uk given u1 hat, u2 hat, uk hat is greater than equal to n times R minus c.

(Refer Slide Time: 40:29)

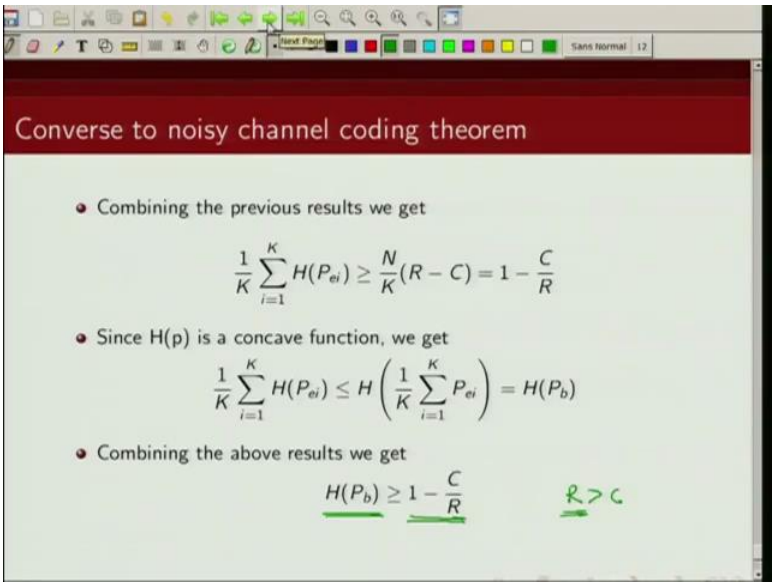


Now, let us further simplify this channel. So, using chain rule i can write this conditional entropy in this particular fashion and as we know, conditioning cannot increase entropy. So, this particular term can be upper bounded by uncertainty in u i given u i hat. So, we

have shown that this particular term is greater than equal to n C and earlier and we are showing that this particular term is greater than this now combining these 2 results we get that h of u i given u i hat sum over all is from 1 to k this is greater than equal to n times R minus c. Now what do we know from Fanos lemma from Fanos Lemma, we know that uncertainty in u i given u hat is less than equal to p e log of l minus 1 where this angular being u i can take l possible values plus h of p e i.

So, this is in this case l is 2 because of binary random variable. So, Fanos Lemma for this will be h of u i given u i hat is less than equal to h of p e i because, the other term p e i log of l minus 1 term that is 0. So, and if you take summation over all i from 1 to k both sides we get this. So, that is what i am saying using Fanos Lemma we get this relation right and. So, on what we proved earlier we showed that this term can is greater than equal to n time's R minus c. So, in other words this particular term is lower bounded by n time's R minus c.

(Refer Slide Time: 43:04)



So, we can write summation of h p e i is lower bounded by n times R minus c, and if you divide both sides by 1 by k. So, divide both side by1 by k what you get is on the right hand side n by k times R minus C which is nothing, but 1 minus capacity divide by the rate next.

This binary entropy function is a concave function and from Jensen's inequality, what do we know about the concave function? So, Jensen's inequality says if the function f of x is concave then expected value of the function is less than equal to function evaluated at expected value. If f of x is concave, so since this binary entropy function is concave expected value of the function here is a binary entropy function should be less than equal to function evaluated at expected value, and what is this term this term is nothing, but average bit error probability.

So, by combining this result, with this result we have shown that binary entropy function of this probability of error is lower bounded by1 by 1 minus C by r, and remember R is greater than C here. So, when R is greater than C this is a non 0 quantity. So, probability of error cannot be 0. So, if we try to transmit at rate adverb capacity our probability of error will be non 0. So, this proofs the converse of the noisy channel coding theorem. So, with this we will conclude our discussion on noisy channel coding theorem.

Thank you.