Outstanding Reading

- Read chapters 1, and 2 in C&T.
- Read chapter 3 in C&T.
- Read section 11.1, 11.3, method of types and universal source coding.
- Read chapter 4.
- Read chapter 5.
- Read stream code chapter 6 in “Information Theory, Inference, and Learning Algorithms” by David J.C. MacKay (available online http://www.inference.phy.cam.ac.uk/mackay/itila/)
- Read chapter 7 in Cover and Thomas, channel capacity
Homework 5 out, due Thursday March 1st, 11:45pm via our dropbox (https://catalyst.uw.edu/collectit/dropbox/karna/19164)

Homework 6 is likely to be available on Friday, March 2nd.

Late policy: 10% every 24 hour period that you are late, and no more than 3 days late accepted.

Lowest grade out of all HW grades is not counted towards final grade (so you can skip one HW with impunity).

Please do use our discussion board (https://catalyst.uw.edu/gopost/board/karna/25503/) for all questions, so that all will benefit from them being answered.
Class Road Map - IT-I

- **L1 (1/3):** Overview, Entropy
- **L2 (1/5):** Props. Entropy, Mutual Information, KL-Divergence
- **L3 (1/10):** KL-Divergence, Jensen, properties, Data Proc. Inequality
- **L4 (1/12):** Data Proc. Ineq., thermodynamics, Stats, Fano
- **L5 (1/17):** Fano, AEP
- **L6 (1/19):** snow
- **L6 (1/24):** AEP, source coding
- **L7 (1/26):** Method of Types
- **L8 (1/31):** Stoc. Proc.
- **L9 (2/2):** HMMs, coding
- **L10 (2/7):** Coding, Kraft,
- **L11 (2/9):** Huffman, midterm
- **L12 (2/14):** Midterm
- **L13 (2/16):** Shannon Games, Arithmetic
- **L14 (2/21):** Channel Capacity
- **L15 (2/23):** Towards Shannon’s Thm.
- **L16 (2/28):** Shannon’s 2nd theorem.
- **L17 (3/1):** Shannon’s 2nd theorem.
- **L18 (3/6):**
- **L19 (3/8):**

Finals Week: March 12th–16th.
Noisy Typewriter

Consider the channel

A \rightarrow A
B \rightarrow B
C \rightarrow C
D \rightarrow D
E \rightarrow E
F \rightarrow F
G \rightarrow G
H \rightarrow H
I \rightarrow I
J \rightarrow J
K \rightarrow K
L \rightarrow L
M \rightarrow M
N \rightarrow N
O \rightarrow O
P \rightarrow P
Q \rightarrow Q
R \rightarrow R
S \rightarrow S
T \rightarrow T
U \rightarrow U
V \rightarrow V
W \rightarrow W
X \rightarrow X
Y \rightarrow Y
Z \rightarrow Z

So 26 input symbols, and each symbol maps probabilistically to itself or its lexicographic neighbor. I.e., 
\( p(A \rightarrow A) = p(A \rightarrow B) = \frac{1}{2} \), etc. Each symbol always has some chance of error, so how can we communicate without error? Choose subset of symbols that can be uniquely disambiguated on receiver side. Choose every other source symbol, A, C, E, etc. Thus A \rightarrow \{A, B\}, C \rightarrow \{C, D\}, E \rightarrow \{E, F\}, etc. so that each received symbols has only one unique source symbol.

Capacity \( C = \log 13 \), visualized on left:

Q: what happens to \( C \) when probabilities are not all \( \frac{1}{2} \)?
Noisy Typewriter

Consider the channel

- So 26 input symbols, and each symbol maps probabilistically to itself or its lexicographic neighbor.
Consider the channel

- So 26 input symbols, and each symbol maps probabilistically to itself or its lexicographic neighbor.
- I.e., \( p(A \rightarrow A) = p(A \rightarrow B) = \frac{1}{2} \), etc.
Noisy Typewriter

Consider the channel

- So 26 input symbols, and each symbol maps probabilistically to itself or its lexicographic neighbor.
- i.e., \( p(A \rightarrow A) = p(A \rightarrow B) = 1/2 \), etc.
- Each symbol always has some chance of error, so how can we communicate without error?
Noisy Typewriter

Consider the channel

- So 26 input symbols, and each symbol maps probabilistically to itself or its lexicographic neighbor.
- I.e., \( p(A \rightarrow A) = p(A \rightarrow B) = \frac{1}{2} \), etc.
- Each symbol always has some chance of error, so how can we communicate without error?
- Choose subset of symbols that can be uniquely disambiguated on receiver side. Choose every other source symbol, A, C, E, etc.
Noisy Typewriter

Consider the channel

- So 26 input symbols, and each symbol maps probabilistically to itself or its lexicographic neighbor.
- I.e., \( p(A \rightarrow A) = p(A \rightarrow B) = 1/2 \), etc.
- Each symbol always has some chance of error, so how can we communicate without error?
- Choose subset of symbols that can be uniquely disambiguated on receiver side. Choose every other source symbol, A, C, E, etc.
- Thus \( A \rightarrow \{A, B\} \), \( C \rightarrow \{C, D\} \), \( E \rightarrow \{E, F\} \), etc. so that each received symbols has only one unique source symbol.
Consider the channel

- So 26 input symbols, and each symbol maps probabilistically to itself or its lexicographic neighbor.
- I.e., \( p(A \rightarrow A) = p(A \rightarrow B) = \frac{1}{2} \), etc.
- Each symbol always has some chance of error, so how can we communicate without error?
- Choose subset of symbols that can be uniquely disambiguated on receiver side. Choose every other source symbol, A, C, E, etc.
- Thus \( A \rightarrow \{A, B\}, C \rightarrow \{C, D\}, E \rightarrow \{E, F\} \), etc. so that each received symbols has only one unique source symbol.
- Capacity \( C = \log 13 \), visualized on left:
Consider the channel

- So 26 input symbols, and each symbol maps probabilistically to itself or its lexicographic neighbor.
- I.e., $p(A \rightarrow A) = p(A \rightarrow B) = 1/2$, etc.
- Each symbol always has some chance of error, so how can we communicate without error?
- Choose subset of symbols that can be uniquely disambiguated on receiver side. Choose every other source symbol, A, C, E, etc.
- Thus $A \rightarrow \{A, B\}$, $C \rightarrow \{C, D\}$, $E \rightarrow \{E, F\}$, etc. so that each received symbols has only one unique source symbol.
- Capacity $C = \log 13$, visualized on left:
- Q: what happens to $C$ when probabilities are not all 1/2?
A bit that is sent will be flipped with probability $p$.  

$p(Y = 1 | X = 0) = 1 - p$ and $p(Y = 0 | X = 0) = p$.

The BSC is an important channel since it is a simple model but at the same time captures some of the difficulties of more complicated channels.

Q: can we still achieve reliable (“guaranteed” error free) communication with this channel?

A: Yes, if $p < 1/2$ and if we do not ask for too high a transmission rate (which would be $R > C$), then we can. Actually, any $p \neq 1/2$ is sufficient.

Intuition: think about AEP and/or block coding.
A bit that is sent will be flipped with probability $p$. 

The Binary Symmetric Channel (BSC) is an important channel since it is a simple model but at the same time captures some of the difficulties of more complicated channels.

Q: can we still achieve reliable (“guaranteed” error free) communication with this channel?

A: Yes, if $p < \frac{1}{2}$ and if we do not ask for too high a transmission rate (which would be $R > C$), then we can. Actually, any $p \neq \frac{1}{2}$ is sufficient.
Binary Symmetric Channel (BSC)

A bit that is sent will be flipped with probability $p$.

- $p(Y = 1|X = 0) = p = 1 - p(Y = 0|X = 0)$. $p(Y = 0|X = 1) = p = p(Y = 1|X = 1)$. 

The BSC is an important channel since it is a simple model but at the same time captures some of the difficulties of more complicated channels.

Q: can we still achieve reliable (“guaranteed” error free) communication with this channel?

A: Yes, if $p < 1/2$ and if we do not ask for too high a transmission rate (which would be $R > C$), then we can. Actually, any $p \neq 1/2$ is sufficient.

Intuition: think about AEP and/or block coding.
**Binary Symmetric Channel (BSC)**

- A bit that is sent will be flipped with probability $p$.
- $p(Y = 1|X = 0) = p = 1 - p(Y = 0|X = 0)$. $p(Y = 0|X = 1) = p = p(Y = 1|X = 1)$.

The BSC is an important channel since it is a simple model but at the same time captures some of the difficulties of more complicated channels.
A bit that is sent will be flipped with probability $p$.

$p(Y = 1|X = 0) = p = 1 - p(Y = 0|X = 0)$. $p(Y = 0|X = 1) = p = p(Y = 1|X = 1)$.

The BSC is an important channel since it is a simple model but at the same time captures some of the difficulties of more complicated channels.

Q: can we still achieve reliable ("guaranteed" error free) communication with this channel?
A bit that is sent will be flipped with probability $p$.

$p(Y = 1|X = 0) = p = 1 - p(Y = 0|X = 0)$. $p(Y = 0|X = 1) = p = p(Y = 1|X = 1)$.

The BSC is an important channel since it is a simple model but at the same time captures some of the difficulties of more complicated channels.

Q: can we still achieve reliable (“guaranteed” error free) communication with this channel? A: Yes, if $p < 1/2$ and if we do not ask for too high a transmission rate (which would be $R > C$), then we can.
A bit that is sent will be flipped with probability $p$.

- $p(Y = 1|X = 0) = p = 1 - p(Y = 0|X = 0)$.
- $p(Y = 0|X = 1) = p = p(Y = 1|X = 1)$.

The BSC is an important channel since it is a simple model but at the same time captures some of the difficulties of more complicated channels.

Q: can we still achieve reliable (“guaranteed” error free) communication with this channel? A: Yes, if $p < 1/2$ and if we do not ask for too high a transmission rate (which would be $R > C$), then we can. Actually, any $p \neq 1/2$ is sufficient.
Binary Symmetric Channel (BSC)

- A bit that is sent will be flipped with probability $p$.
- $p(Y = 1|X = 0) = p = 1 - p(Y = 0|X = 0)$. $p(Y = 0|X = 1) = p = p(Y = 1|X = 1)$.

-The BSC is an important channel since it is a simple model but at the same time captures some of the difficulties of more complicated channels.

Q: can we still achieve reliable (“guaranteed” error free) communication with this channel? A: Yes, if $p < 1/2$ and if we do not ask for too high a transmission rate (which would be $R > C$), then we can. Actually, any $p \neq 1/2$ is sufficient.

Intuition: think about AEP and/or block coding.
**Binary Symmetric Channel (BSC)**

- A bit that is sent will be flipped with probability $p$.
- $p(Y = 1|X = 0) = p = 1 - p(Y = 0|X = 0)$. $p(Y = 0|X = 1) = p = p(Y = 1|X = 1)$.

The BSC is an important channel since it is a simple model but at the same time captures some of the difficulties of more complicated channels.

Q: can we still achieve reliable ("guaranteed" error free) communication with this channel? A: Yes, if $p < 1/2$ and if we do not ask for too high a transmission rate (which would be $R > C$), then we can. Actually, any $p \neq 1/2$ is sufficient.

Intuition: think about AEP and/or block coding.

But how to compute $C$ the capacity?
BSC Capacity

\[
I(X; Y) = H(Y) - H(Y|X)
\]
BSC Capacity

\[ I(X; Y) = H(Y) - H(Y|X) = H(Y) - \sum_x p(x)H(Y|X = x) \] (1)

(2)
BSC Capacity

\[
I(X; Y) = H(Y) - H(Y|X) = H(Y) - \sum_x p(x)H(Y|X = x) \quad (1)
\]

\[
= H(Y) - \sum_x p(x)H(p) \quad (2)
\]
BSC Capacity

\[
I(X;Y) = H(Y) - H(Y|X) = H(Y) - \sum_x p(x)H(Y|X = x) \tag{1}
\]

\[
= H(Y) - \sum_x p(x)H(p) = H(Y) - H(p) \tag{2}
\]
BSC Capacity

\[ I(X; Y) = H(Y) - H(Y|X) = H(Y) - \sum_x p(x) H(Y|X = x) \quad (1) \]

\[ = H(Y) - \sum_x p(x) H(p) = H(Y) - H(p) \leq 1 - H(p) \quad (2) \]
BSC Capacity

\[ I(X; Y) = H(Y) - H(Y|X) = H(Y) - \sum_x p(x)H(Y|X = x) \]  

\[ = H(Y) - \sum_x p(x)H(p) = H(Y) - H(p) \leq 1 - H(p) \]  

When is \( H(Y) = 1 \)? Note that

\[ \Pr(Y = 1) \]
### BSC Capacity

![Diagram of a binary symmetric channel](image)

$I(X; Y) = H(Y) - H(Y|X) = H(Y) - \sum_x p(x)H(Y|X = x)$ \hspace{1cm} (1)

$$= H(Y) - \sum_x p(x)H(p) = H(Y) - H(p) \leq 1 - H(p)$$ \hspace{1cm} (2)

- **When is $H(Y) = 1$?** Note that

$$\Pr(Y = 1) = \Pr(Y = 1|X = 1)\Pr(X = 1)$$ \hspace{1cm} (3)

$$+ \Pr(Y = 1|X = 0)\Pr(X = 0)$$ \hspace{1cm} (4)

$$= (1-p)\Pr(X = 1) + p(1-p) = (1-p)p + p(1-p) = (1-p)p + p(1-p)$$ \hspace{1cm} (5)

$$= (1-p)p + p(1-p) = (1-p^2)p$$ \hspace{1cm} (6)
BSC Capacity

\[
I(X; Y) = H(Y) - H(Y|X) = H(Y) - \sum_x p(x)H(Y|X = x) \tag{1}
\]

\[
= H(Y) - \sum_x p(x)H(p) = H(Y) - H(p) \leq 1 - H(p) \tag{2}
\]

- When is \(H(Y) = 1\)? Note that

\[
\Pr(Y = 1) = \Pr(Y = 1|X = 1)\Pr(X = 1) + \Pr(Y = 1|X = 0)\Pr(X = 0) \tag{3}
\]

\[
= (1 - p)\Pr(X = 1) + p\Pr(X = 0) \tag{4}
\]
BSC Capacity

\[
I(X; Y) = H(Y) - H(Y|X) = H(Y) - \sum_x p(x) H(Y|X = x) \quad (1)
\]

\[
= H(Y) - \sum_x p(x) H(p) = H(Y) - H(p) \leq 1 - H(p) \quad (2)
\]

- When is \( H(Y) = 1 \)? Note that

\[
\Pr(Y = 1) = \Pr(Y = 1|X = 1)\Pr(X = 1) \quad (3)
\]

\[
+ \Pr(Y = 1|X = 0)\Pr(X = 0) \quad (4)
\]

\[
= (1 - p)\Pr(X = 1) + p\Pr(X = 0) \quad (5)
\]

\[
= (1 - p)\Pr(X = 1) + p(1 - \Pr(X = 1)) \quad (6)
\]
BSC Capacity

\[
I(X; Y) = H(Y) - H(Y|X) = H(Y) - \sum_x p(x)H(Y|X = x)
\]

\[
= H(Y) - \sum_x p(x)H(p) = H(Y) - H(p) \leq 1 - H(p)
\]

**When is** \( H(Y) = 1? \)** Note that**

\[
\Pr(Y = 1) = \Pr(Y = 1|X = 1)\Pr(X = 1) + \Pr(Y = 1|X = 0)\Pr(X = 0)
\]

\[
= (1 - p)\Pr(X = 1) + p\Pr(X = 0)
\]

\[
= (1 - p)\Pr(X = 1) + p(1 - \Pr(X = 1))
\]

**So** \( H(Y) = 1 \) **if** \( H(X) = 1 \) **(i.e.,** \( \Pr(X = 1) = 1/2) \)
BSC Capacity

\[
I(X; Y) = H(Y) - H(Y|X) = H(Y) - \sum_x p(x)H(Y|X = x) \quad (1)
\]

\[
= H(Y) - \sum_x p(x)H(p) = H(Y) - H(p) \leq 1 - H(p) \quad (2)
\]

- When is \( H(Y) = 1 \)? Note that

\[
\Pr(Y = 1) = \Pr(Y = 1|X = 1)\Pr(X = 1) \quad (3)
\]

\[
+ \Pr(Y = 1|X = 0)\Pr(X = 0) \quad (4)
\]

\[
= (1 - p)\Pr(X = 1) + p\Pr(X = 0) \quad (5)
\]

\[
= (1 - p)\Pr(X = 1) + p(1 - \Pr(X = 1)) \quad (6)
\]

- So \( H(Y) = 1 \) if \( H(X) = 1 \) (i.e., \( \Pr(X = 1) = 1/2 \))

- Thus, we get that \( C = 1 - H(p) \) which happens when \( X \) is uniform.
Thus, we get that $C = 1 - H(p)$ which happens when $X$ is uniform.
Thus, we get that $C = 1 - H(p)$ which happens when $X$ is uniform.

If $p = 1/2$ then $C = 0$, so if it randomly flips bits, then no information can be sent.
Thus, we get that $C = 1 - H(p)$ which happens when $X$ is uniform.

If $p = 1/2$ then $C = 0$, so if it randomly flips bits, then no information can be sent.

If $p \neq 1/2$, then we can communicate, albeit potentially slowly. E.g., if $p = 0.499$ then $C = 2.8854 \times 10^{-6}$ bits per channel use. So to send one bit, need to use the channel quite a bit.
Thus, we get that \( C = 1 - H(p) \) which happens when \( X \) is uniform.

If \( p = 1/2 \) then \( C = 0 \), so if it randomly flips bits, then no information can be sent.

If \( p \neq 1/2 \), then we can communicate, albeit potentially slowly. E.g., if \( p = 0.499 \) then \( C = 2.8854 \times 10^{-6} \) bits per channel use. So to send one bit, need to use the channel quite a bit.

If \( p = 0 \) or \( p = 1 \), then \( C = 1 \) and we can get maximum possible rate (i.e., the capacity is one bit per channel use).

Thus, we get that \( C = 1 - H(p) \) which happens when \( X \) is uniform.
Decoding

- Lets temporarily look ahead to address this problem.
Let’s temporarily look ahead to address this problem.

We can “decode” the source using the received string, source distribution, and the channel model \( p(y|x) \) via Bayes rule. I.e.

\[
Pr(x|y) = \frac{\text{channel} \cdot \text{source}}{Pr(y)} = \frac{Pr(y|x)Pr(x)}{Pr(y)} = \frac{Pr(y|x)Pr(x)}{\sum_{x'} Pr(y|x')Pr(x')}
\]  

(7)
Decoding

- Lets temporarily look ahead to address this problem.

- We can “decode” the source using the received string, source distribution, and the channel model \( p(y|x) \) via Bayes rule. I.e.

\[
Pr(x|y) = \frac{Pr(y|x) Pr(x)}{Pr(y)} = \frac{Pr(y|x) Pr(x)}{\sum_{x'} Pr(y|x') Pr(x')}
\]

(7)

- If we get a particular \( y \), we can compute \( p(x|y) \) and make a decision based on that. I.e., \( \hat{x} = \text{argmax}_x p(x|y) \) (maximum likelihood decoding).
Decoding

- Lets temporarily look ahead to address this problem.
- We can “decode” the source using the received string, source distribution, and the channel model $p(y|x)$ via Bayes rule. I.e.

$$
\Pr(x|y) = \frac{\Pr(y|x) \Pr(x)}{\Pr(y)} = \frac{\Pr(y|x) \Pr(x)}{\sum_{x'} \Pr(y|x') \Pr(x')}
$$

If we get a particular $y$, we can compute $p(x|y)$ and make a decision based on that. I.e., $\hat{x} = \arg\max_x p(x|y)$ (maximum likelihood decoding).
- This is optimal decoding in that it minimizes the error.
Decoding

- Lets temporarily look ahead to address this problem.
- We can “decode” the source using the received string, source distribution, and the channel model $p(y|x)$ via Bayes rule. I.e.

$$
\Pr(x|y) = \frac{\text{channel} \cdot \text{source}}{\Pr(y)} = \frac{\Pr(y|x)\Pr(x)}{\sum_{x'} \Pr(y|x')\Pr(x')}
$$

(7)

- If we get a particular $y$, we can compute $p(x|y)$ and make a decision based on that. I.e., $\hat{x} = \arg\max_x p(x|y)$ (maximum likelihood decoding).
- This is optimal decoding in that it minimizes the error.
- Error if $x \neq \hat{x}$, and $\Pr(\text{error}) = \Pr(x \neq \hat{x})$. 
Decoding

- Lets temporarily look ahead to address this problem.
- We can “decode” the source using the received string, source distribution, and the channel model $p(y|x)$ via Bayes rule. I.e.

$$
Pr(x|y) = \frac{\text{channel} \cdot \text{source}}{Pr(y)} = \frac{Pr(y|x)Pr(x)}{Pr(y)} = \frac{Pr(y|x)Pr(x)}{\sum_{x'} Pr(y|x')Pr(x')}
$$

(7)

- If we get a particular $y$, we can compute $p(x|y)$ and make a decision based on that. I.e., $\hat{x} = \arg\max_x p(x|y)$ (maximum likelihood decoding).

- This is optimal decoding in that it minimizes the error.

- Error if $x \neq \hat{x}$, and $Pr(\text{error}) = Pr(x \neq \hat{x})$.

- This is minimal if we chose $\arg\max_x p(x|y)$ since the error $1 - Pr(\hat{x}|y)$ is minimal.
Properties of Channel Capacity $C$

- $C \geq 0$ since $I(X;Y) \geq 0$. 

Interestingly, since concave, this makes computing something like the capacity easier. I.e., a local maximum is a global maximum, and computing the capacity for a general channel model is a convex optimization procedure.
Properties of Channel Capacity $C$

- $C \geq 0$ since $I(X; Y) \geq 0$.
- $C \leq \log |\mathcal{X}|$ since $C = \max_p(I(X; Y)) \leq \max H(X) = \log |\mathcal{X}|$. 
Properties of Channel Capacity $C$

- $C \geq 0$ since $I(X; Y) \geq 0$.
- $C \leq \log |X|$ since $C = \max_{p(x)} I(X; Y) \leq \max H(X) = \log |X|$.
- $C \leq \log |Y|$ for same reason.
Properties of Channel Capacity $C$

- $C \geq 0$ since $I(X; Y) \geq 0$.
- $C \leq \log |\mathcal{X}|$ since $C = \max_{p(x)} I(X; Y) \leq \max H(X) = \log |\mathcal{X}|$.
- $C \leq \log |\mathcal{Y}|$ for same reason. Thus, the alphabet sizes limit the transmission rate.
Properties of Channel Capacity $C$

- $C \geq 0$ since $I(X; Y) \geq 0$.
- $C \leq \log |\mathcal{X}|$ since $C = \max_{p(x)} I(X; Y) \leq \max H(X) = \log |\mathcal{X}|$.
- $C \leq \log |\mathcal{Y}|$ for same reason. Thus, the alphabet sizes limit the transmission rate.
- $I(X; Y) = I_{p(x)}(X; Y)$ is a continuous function of $p(x)$.
Properties of Channel Capacity $C$

- $C \geq 0$ since $I(X; Y) \geq 0$.
- $C \leq \log |X|$ since $C = \max_{p(x)} I(X; Y) \leq \max H(X) = \log |X|$.
- $C \leq \log |Y|$ for same reason. Thus, the alphabet sizes limit the transmission rate.
- $I(X; Y) = I_{p(x)}(X; Y)$ is a continuous function of $p(x)$.
- Recall, $I(X; Y)$ is a concave function of $p(x)$ for fixed $p(y|x)$. 


Properties of Channel Capacity $C$

- $C \geq 0$ since $I(X; Y) \geq 0$.
- $C \leq \log |\mathcal{X}|$ since $C = \max_{p(x)} I(X; Y) \leq \max H(X) = \log |\mathcal{X}|$.
- $C \leq \log |\mathcal{Y}|$ for same reason. Thus, the alphabet sizes limit the transmission rate.

$I(X; Y) = I_{p(x)}(X; Y)$ is a continuous function of $p(x)$.

Recall, $I(X; Y)$ is a concave function of $p(x)$ for fixed $p(y|x)$.

Thus, $I_{\lambda p_1 + (1-\lambda)p_2}(X; Y) \geq \lambda I_{p_1}(X; Y) + (1-\lambda)I_{p_2}(X; Y)$. 
Properties of Channel Capacity $C$

- $C \geq 0$ since $I(X;Y) \geq 0$.
- $C \leq \log |\mathcal{X}|$ since $C = \max_{p(x)} I(X;Y) \leq \max H(X) = \log |\mathcal{X}|$.
- $C \leq \log |\mathcal{Y}|$ for same reason. Thus, the alphabet sizes limit the transmission rate.
- $I(X;Y) = I_{p(x)}(X;Y)$ is a continuous function of $p(x)$.
- Recall, $I(X;Y)$ is a concave function of $p(x)$ for fixed $p(y|x)$. Thus, $I_{\lambda p_1 + (1-\lambda)p_2}(X;Y) \geq \lambda I_{p_1}(X;Y) + (1-\lambda)I_{p_2}(X;Y)$.
- Interestingly, since concave, this makes computing something like the capacity easier. I.e., a local maximum is a global maximum, and computing the capacity for a general channel model is a convex optimization procedure.
Properties of Channel Capacity $C$

- $C \geq 0$ since $I(X;Y) \geq 0$.
- $C \leq \log |\mathcal{X}|$ since $C = \max_{p(x)} I(X;Y) \leq \max H(X) = \log |\mathcal{X}|$.
- $C \leq \log |\mathcal{Y}|$ for same reason. Thus, the alphabet sizes limit the transmission rate.
- $I(X;Y) = I_{p(x)}(X;Y)$ is a continuous function of $p(x)$.
- Recall, $I(X;Y)$ is a concave function of $p(x)$ for fixed $p(y|x)$. Thus, $I_{\lambda p_1 + (1-\lambda)p_2}(X;Y) \geq \lambda I_{p_1}(X;Y) + (1-\lambda)I_{p_2}(X;Y)$.
- Interestingly, since concave, this makes computing something like the capacity easier. I.e., a local maximum is a global maximum, and computing the capacity for a general channel model is a convex optimization procedure.
- Recall also, $I(X;Y)$ is a convex function of $p(y|x)$ for fixed $p(x)$. 
One of the most important theorems of the last century.
Shannon’s 2nd Theorem

- One of the most important theorems of the last century.
- We’ll see it in various forms, but we state it here somewhat informally to start acquiring intuition.
Shannon’s 2nd Theorem

- One of the most important theorems of the last century.
- We’ll see it in various forms, but we state it here somewhat informally to start acquiring intuition.
Shannon’s 2nd Theorem

- One of the most important theorems of the last century.
- We’ll see it in various forms, but we state it here somewhat informally to start acquiring intuition.

**Theorem 2.1 (Shannon’s 2nd Theorem)**

$C$ is the maximum number of bits (on average, per channel use) that we can transmit over a channel reliably.
Shannon’s 2nd Theorem

- One of the most important theorems of the last century.
- We’ll see it in various forms, but we state it here somewhat informally to start acquiring intuition.

**Theorem 2.1 (Shannon’s 2nd Theorem)**

\[ C \] is the maximum number of bits (on average, per channel use) that we can transmit over a channel reliably.

- Here, “reliably” means with vanishingly small and exponentially decreasing probability of error as the block length gets longer. We can easily make this probability essentially zero.
Shannon’s 2nd Theorem

- One of the most important theorems of the last century.
- We’ll see it in various forms, but we state it here somewhat informally to start acquiring intuition.

Theorem 2.1 (Shannon’s 2nd Theorem)

\[ C \] is the maximum number of bits (on average, per channel use) that we can transmit over a channel reliably.

- Here, “reliably” means with vanishingly small and exponentially decreasing probability of error as the block length gets longer. We can easily make this probability essentially zero.
- Conversely, if we try to push \( > C \) bits through the channel, error quickly goes to 1.
Shannon’s 2nd Theorem

- Intuition of this we’ve already seen in the noisy typewriter and the region partitioning.
Shannon’s 2nd Theorem

- Intuition of this we’ve already seen in the noisy typewriter and the region partitioning.
- Slightly more precisely, this is a sort of bin packing problem.
Shannon’s 2nd Theorem

- Intuition of this we’ve already seen in the noisy typewriter and the region partitioning.
- Slightly more precisely, this is a sort of bin packing problem.
- We’ve got a region of possible codewords, and we pack as many smaller non-overlapping bins into the region as possible.
Shannon’s 2nd Theorem

- Intuition of this we’ve already seen in the noisy typewriter and the region partitioning.
- Slightly more precisely, this is a sort of bin packing problem.
- We’ve got a region of possible codewords, and we pack as many smaller non-overlapping bins into the region as possible.
- The smaller bins correspond to the noise in the channel, and the packing problem depends on the underlying “shape”
Shannon’s 2nd Theorem

- Intuition of this we’ve already seen in the noisy typewriter and the region partitioning.
- Slightly more precisely, this is a sort of bin packing problem.
- We’ve got a region of possible codewords, and we pack as many smaller non-overlapping bins into the region as possible.
- The smaller bins correspond to the noise in the channel, and the packing problem depends on the underlying “shape”
- Not really a partition, since there might be wasted space, also depending on the bin and region shapes.
Shannon’s 2nd Theorem

- Intuition of this we’ve already seen in the noisy typewriter and the region partitioning.
- Slightly more precisely, this is a sort of bin packing problem.
- We’ve got a region of possible codewords, and we pack as many smaller non-overlapping bins into the region as possible.
- The smaller bins correspond to the noise in the channel, and the packing problem depends on the underlying “shape”
- Not really a partition, since there might be wasted space, also depending on the bin and region shapes.
Shannon’s 2nd Theorem

- Intuitive idea: use typicality argument, like in chapter 3.
Shannon’s 2nd Theorem

- Intuitive idea: use typicality argument, like in chapter 3.
- There are $\approx 2^{nH(X)}$ typical sequences, each with probability $2^{-nH(X)}$ and with $p(A_{\epsilon}^{(n)}) \approx 1$, so the only thing with “any” probability is the typical set and it has all the probability.
Shannon’s 2nd Theorem

- Intuitive idea: use typicality argument, like in chapter 3.
- There are $\approx 2^{nH(X)}$ typical sequences, each with probability $2^{-nH(X)}$ and with $p(A_{\epsilon}^{(n)}) \approx 1$, so the only thing with “any” probability is the typical set and it has all the probability.
- The same thing is true for conditional entropy.
Shannon’s 2nd Theorem

- Intuitive idea: use typicality argument, like in chapter 3.
- There are \( \approx 2^{nH(X)} \) typical sequences, each with probability \( 2^{-nH(X)} \) and with \( p(A^{(n)}_\epsilon) \approx 1 \), so the only thing with “any” probability is the typical set and it has all the probability.
- The same thing is true for conditional entropy.
- That is, for a typical input \( X \), there are \( \approx 2^{nH(Y|X)} \) output sequences.
Intuitive idea: use typicality argument, like in chapter 3.

There are \( \approx 2^{nH(X)} \) typical sequences, each with probability \( 2^{-nH(X)} \) and with \( p(A^X_n) \approx 1 \), so the only thing with “any” probability is the typical set and it has all the probability.

The same thing is true for conditional entropy.

That is, for a typical input \( X \), there are \( \approx 2^{nH(Y|X)} \) output sequences.

Overall, there are \( 2^{nH(Y)} \) typical output sequences, and we know that \( 2^{nH(Y)} \geq 2^{nH(Y|X)} \).
Shannon’s 2nd Theorem: Intuition

- Goal: find a non-confusable subset of the inputs that produce disjoint output sequences (as in picture).

\[ \approx 2^n H(Y) \text{ (typical) outputs (i.e., the marginally typical } Y \text{ sequences).} \]
\[ \approx 2^n H(Y|X) \text{ (} X \text{-conditionally typical } Y \text{ sequences).} \]
\[ \equiv \text{ the average possible number of outputs for a possible input, so this many could be confused with each other.} \]
\[ \text{I.e., on average, for a given } X = x, \text{ this is approximately how many outputs there might be.} \]

\[ \text{So the number of non-confusable inputs is } \leq 2^n H(Y) 2^n H(Y|X) = 2^n (H(Y) - H(Y|X)) = 2^n I(X;Y) \] (8)

Note, in non-ideal case, there could be overlap of the typical \( Y \)-given-\( X \) sequences, but the best we can do (in terms of maximizing the number of non-confusable inputs) is when there is no overlap on the output. This is assumed in the above.
Shannon’s 2nd Theorem: Intuition

- Goal: find a non-confusable subset of the inputs that produce disjoint output sequences (as in picture).
- There are $\approx 2^{nH(Y)}$ (typical) outputs (i.e., the marginally typical $Y$ sequences).
Shannon’s 2nd Theorem: Intuition

- Goal: find a non-confusable subset of the inputs that produce disjoint output sequences (as in picture).
- There are $\approx 2^{nH(Y)}$ (typical) outputs (i.e., the marginally typical $Y$ sequences).
- There are $\approx 2^{nH(Y|X)}$ ($X$-conditionally typical $Y$ sequences) outputs.
Shannon’s 2nd Theorem: Intuition

- Goal: find a non-confusable subset of the inputs that produce disjoint output sequences (as in picture).
- There are \( \approx 2^{nH(Y)} \) (typical) outputs (i.e., the marginally typical \( Y \) sequences).
- There are \( \approx 2^{nH(Y|X)} \) (\( X \)-conditionally typical \( Y \) sequences) outputs. \( \equiv \) the average possible number of outputs for a possible input, so this many could be confused with each other.
Shannon’s 2nd Theorem: Intuition

- Goal: find a non-confusable subset of the inputs that produce disjoint output sequences (as in picture).
- There are \( \approx 2^{nH(Y)} \) (typical) outputs (i.e., the marginally typical \( Y \) sequences).
- There are \( \approx 2^{nH(Y|X)} \) (\( X \)-conditionally typical \( Y \) sequences) outputs. \( \equiv \) the average possible number of outputs for a possible input, so this many could be confused with each other. I.e., on average, for a given \( X = x \), this is approximately how many outputs there might be.
Shannon’s 2nd Theorem: Intuition

- Goal: find a non-confusable subset of the inputs that produce disjoint output sequences (as in picture).
- There are \( \approx 2^{nH(Y)} \) (typical) outputs (i.e., the marginally typical \( Y \) sequences).
- There are \( \approx 2^{nH(Y|X)} \) (\( X \)-conditionally typical \( Y \) sequences) outputs. \( \equiv \) the average possible number of outputs for a possible input, so this many could be confused with each other. I.e., on average, for a given \( X = x \), this is approximately how many outputs there might be.
- So the number of non-confusable inputs is

\[
\leq \frac{2^{nH(Y)}}{2^{nH(Y|X)}} = 2^{n(H(Y) - H(Y|X))} = 2^{nI(X;Y)} \quad (8)
\]
Shannon’s 2nd Theorem: Intuition

- **Goal:** find a non-confusable subset of the inputs that produce disjoint output sequences (as in picture).
- There are \( \approx 2^{nH(Y)} \) (typical) outputs (i.e., the marginally typical \( Y \) sequences).
- There are \( \approx 2^{nH(Y|X)} \) (\( X \)-conditionally typical \( Y \) sequences) outputs. \( \equiv \) the average possible number of outputs for a possible input, so this many could be confused with each other. I.e., on average, for a given \( X = x \), this is approximately how many outputs there might be.
- So the number of non-confusable inputs is
  \[
  \leq \frac{2^{nH(Y)}}{2^{nH(Y|X)}} = 2^{n(H(Y) - H(Y|X))} = 2^{nI(X;Y)} \quad (8)
  \]
- Note, in non-ideal case, there could be overlap of the typical \( Y \)-given-\( X \) sequences, but the best we can do (in terms of maximizing the number of non-confusable inputs) is when there is no overlap on the output. This is assumed in the above.
Shannon’s 2nd Theorem: Intuition

- The number of non-confusable inputs is

\[ \leq \frac{2^n H(Y)}{2^n H(Y|X)} = 2^n (H(Y) - H(Y|X)) = 2^n I(X;Y) \] (9)
Shannon’s 2nd Theorem: Intuition

- The number of non-confusable inputs is

\[ \frac{2^n H(Y)}{2^n H(Y|X)} \leq 2^n (H(Y) - H(Y|X)) = 2^n I(X;Y) \]  

- We can view this as a volume. \(2^n H(Y)\) is the total number of possible slots, while \(2^n H(Y|X)\) is the number of slots taken up (on average) for a given source word. Thus, the number of source words that can be used is the ratio.
Shannon’s 2nd Theorem: Intuition

- The number of non-confusable inputs is

\[
\leq \frac{2^n H(Y)}{2^n H(Y|X)} = 2^n (H(Y) - H(Y|X)) = 2^n I(X;Y)
\]  

(9)

- We can view this as a volume. $2^n H(Y)$ is the total number of possible slots, while $2^n H(Y|X)$ is the number of slots taken up (on average) for a given source word. Thus, the number of source words that can be used is the ratio.
Shannon’s 2nd Theorem: Intuition

The number of non-confusable inputs is

\[
\leq \frac{2^n H(Y)}{2^n H(Y|X)} = 2^n (H(Y) - H(Y|X)) = 2^n I(X;Y)
\]  

(10)
Shannon’s 2nd Theorem: Intuition

- The number of non-confusable inputs is

\[
\leq \frac{2^n H(Y)}{2^n H(Y|X)} = 2^n (H(Y) - H(Y|X)) = 2^n I(X;Y) \tag{10}
\]

- Now of course, to maximize this number, for a fixed channel \( p(y|x) \), we find the best \( p(x) \) which gives \( I(X;Y) = C \), which is the log of the maximum number of inputs possible to use.
Shannon’s 2nd Theorem: Intuition

- The number of non-confusable inputs is

\[
\leq \frac{2^n H(Y)}{2^n H(Y|X)} = 2^n (H(Y) - H(Y|X)) = 2^n I(X;Y) \tag{10}
\]

- Now of course, to maximize this number, for a fixed channel \(p(y|x)\), we find the best \(p(x)\) which gives \(I(X;Y) = C\), which is the log of the maximum number of inputs possible to use.

- This is the capacity.
Some Definitions

- Reminder: model of communication: noise $p(y|x)$

\[ W \in \{1, \ldots, M\} \text{ requiring } \log M \text{ bits per message.} \]

Signal sent through channel $X^n(W)$, a random codeword. Received signal from channel $Y^n \sim p(y^n|x^n)$. Decoding via guess $\hat{W} = g(Y^n)$.

Discrete memoryless channel (DMC) $(X^n, p(y^n|x^n), Y^n)$ $n$th extension to channel is $(X^n, p(y^n|x^n), Y^n)$. Feedback if $x_k$ can use both previous inputs and outputs. No feedback if $p(x_k|x_1:k-1, y_1:k-1) = p(x_k|x_1:k-1)$. We'll analyze feedback a bit later.
Some Definitions

- Reminder: model of communication:
  - **noise** $p(y|x)$

```
source -> encoder -> channel -> decoder -> receiver
```

- **Message** $W \in \{1, \ldots, M\}$ requiring $\log M$ bits per message.
Some Definitions

- Reminder: model of communication:
  - **noise** \( p(y|x) \)

  \[
  \begin{align*}
  \text{source} & \rightarrow \text{encoder} & \rightarrow \text{channel} & \rightarrow \text{decoder} & \rightarrow \text{receiver} \\
  W & \rightarrow X^n & \rightarrow Y^n & \rightarrow \hat{W} \\
  n = \log M \text{ bits} & \quad n \log |X| \text{ bits} & \quad n \log |Y| \text{ bits} \\
  \end{align*}
  \]

- **Message** \( W \in \{1, \ldots, M\} \) requiring \( \log M \) bits per message.
- **Signal** sent through channel \( X^n(W) \), a random codeword.
Some Definitions

- Reminder: model of communication:
  
  \[ \text{noise } p(y|x) \]

\begin{center}
\begin{tikzpicture}
  \node (source) {source};
  \node (encoder) [right of=source] {encoder};
  \node (channel) [right of=encoder] {channel};
  \node (decoder) [right of=channel] {decoder};
  \node (receiver) [right of=decoder] {receiver};

  \draw [->] (source) -- (encoder);
  \draw [->] (encoder) -- (channel);
  \draw [->] (channel) -- (decoder);
  \draw [->] (decoder) -- (receiver);

  \node [above of=source] {$W$};
  \node [above of=encoder] {$X^n$};
  \node [above of=channel] {$Y^n$};
  \node [above of=receiver] {$\hat{W}$};

  \node [left of=encoder, node distance=2cm] {$n = \log M \text{ bits}$};
  \node [right of=channel, node distance=2cm] {$n \log |X| \text{ bits}$};
  \node [right of=receiver, node distance=2cm] {$n \log |Y| \text{ bits}$};
\end{tikzpicture}
\end{center}

- **Message** \( W \in \{1, \ldots, M\} \) requiring \( \log M \) bits per message.
- **Signal** sent through channel \( X^n(W) \), a random codeword.
- **Received signal** from channel \( Y^n \sim p(y^n|x^n) \)
Some Definitions

- Reminder: model of communication: **noise** $p(y|x)$

    ![Diagram]

- **Message** $W \in \{1, \ldots, M\}$ requiring $\log M$ bits per message.
- **Signal** sent through channel $X^n(W)$, a random codeword.
- **Received signal** from channel $Y^n \sim p(y^n|x^n)$
- **Decoding** via guess $\hat{W} = g(Y^n)$. 

Reminder: model of communication:

- Source
- Encoder
- Channel
- Decoder
- Receiver

$W \in \{1, \ldots, M\}$ requiring $\log M$ bits per message.
Some Definitions

- **Reminder:** model of communication:
  \[ \text{noise } p(y|x) \]

- **Message** \( W \in \{1, \ldots, M\} \) requiring \( \log M \) bits per message.
- **Signal** sent through channel \( X^n(W) \), a random codeword.
- **Received signal** from channel \( Y^n \sim p(y^n|x^n) \)
- **Decoding** via guess \( \hat{W} = g(Y^n) \).
- **Discrete memoryless channel (DMC)** \( (X, p(y|x), Y) \)
Some Definitions

- Reminder: model of communication:
  - **noise** $p(y|x)$
  - Source
  - Encoder
  - Channel
  - Decoder
  - Receiver
  - $W$ $\in \{1, \ldots, M\}$ requiring $\log M$ bits per message.
  - Signal sent through channel $X^n(W)$, a random codeword.
  - Received signal from channel $Y^n \sim p(y^n|x^n)$
  - Decoding via guess $\hat{W} = g(Y^n)$.
  - Discrete memoryless channel (DMC) $(\mathcal{X}, p(y|x), \mathcal{Y})$
  - $n^{th}$ extension to channel is $(\mathcal{X}^n, p(y^n|x^n), \mathcal{Y}^n)$
### Some Definitions

- **Reminder:** model of communication: \( \text{noise } p(y|x) \)

![Diagram showing the model of communication: source, encoder, channel, decoder, receiver.]

- \( W \) bits
- \( X^n \) bits
- \( Y^n \) bits
- \( \hat{W} \)

- **Message** \( W \in \{1, \ldots, M\} \) requiring \( \log M \) bits per message.
- **Signal** sent through channel \( X^n(W) \), a random codeword.
- **Received signal** from channel \( Y^n \sim p(y^n|x^n) \)
- **Decoding** via guess \( \hat{W} = g(Y^n) \).
- **Discrete memoryless channel (DMC)** \( (\mathcal{X}, p(y|x), \mathcal{Y}) \)
- \( n^{th} \) extension to channel is \( (\mathcal{X}^n, p(y^n|x^n), \mathcal{Y}^n) \)
- **Feedback** if \( x_k \) can use both previous inputs and outputs.
Some Definitions

- Reminder: model of communication:
  \[ p(y|x) \]

![Diagram of communication model]

- **Message** \( W \in \{1, \ldots, M\} \) requiring \( \log M \) bits per message.
- **Signal** sent through channel \( X^n(W) \), a random codeword.
- **Received signal** from channel \( Y^n \sim p(y^n|x^n) \)
- **Decoding** via guess \( \hat{W} = g(Y^n) \).
- **Discrete memoryless channel (DMC)** \( (\mathcal{X}, p(y|x), \mathcal{Y}) \)
- \( n^{th} \) extension to channel is \( (\mathcal{X}^n, p(y^n|x^n), \mathcal{Y}^n) \)
- **Feedback** if \( x_k \) can use both previous inputs and outputs.
- **No feedback** if \( p(x_k|x_{1:k-1}, y_{1:k-1}) = p(x_k|x_{1:k-1}) \). We’ll analyze feedback a bit later.
Definition 2.2 \((M, n)\) code

An \((M, n)\) code for channel \((\mathcal{X}, p(y|x), \mathcal{Y})\) is:

1. An index set \(\{1, 2, \ldots, M\}\)
2. An encoding function \(X^n : \{1, 2, \ldots, M\} \rightarrow X^n\) yielding codewords \(X^n(1), X^n(2), X^n(3), \ldots, X^n(M)\).

Each source message has a codeword, and each codeword is \(n\) code symbols.
3. A decoding function, i.e., \(g : Y^n \rightarrow \{1, 2, \ldots, M\}\) which makes a "guess" about original message given channel output.

In an \((M, n)\) code, \(M = \) the number of possible messages to be sent, and \(n = \) number of channel uses by the codewords of the code.
Definition 2.2 \((M, n)\) code

An \((M, n)\) code for channel \((\mathcal{X}, p(y|x), \mathcal{Y})\) is:

1. An index set \(\{1, 2, \ldots, M\}\)
**Definition 2.2 ((M, n) code)**

An \((M, n)\) code for channel \((\mathcal{X}, p(y|x), \mathcal{Y})\) is:

1. An index set \(\{1, 2, \ldots, M\}\)
2. An encoding function \(X^n : \{1, 2, \ldots, M\} \rightarrow \mathcal{X}^n\) yielding codewords \(X^n(1), X^n(2), X^n(3), \ldots, X^n(M)\).
Definition 2.2 \((M, n)\) code

An \((M, n)\) code for channel \((\mathcal{X}, p(y|x), \mathcal{Y})\) is:

1. An index set \(\{1, 2, \ldots, M\}\)
2. An encoding function \(X^n : \{1, 2, \ldots, M\} \rightarrow \mathcal{X}^n\) yielding codewords \(X^n(1), X^n(2), X^n(3), \ldots, X^n(M)\). Each source message has a codeword, and each codeword is \(n\) code symbols.
Definition 2.2 \((M, n)\) code

An \((M, n)\) code for channel \((\mathcal{X}, p(y|x), \mathcal{Y})\) is:

1. An index set \(\{1, 2, \ldots, M\}\)

2. An encoding function \(X^n : \{1, 2, \ldots, M\} \rightarrow \mathcal{X}^n\) yielding codewords \(X^n(1), X^n(2), X^n(3), \ldots, X^n(M)\). Each source message has a codeword, and each codeword is \(n\) code symbols.

3. Decoding function, i.e., \(g : \mathcal{Y}^n \rightarrow \{1, 2, \ldots, M\}\) which makes a “guess” about original message given channel output.
\((M, n)\) code

**Definition 2.2 \((M, n)\) code**

An \((M, n)\) code for channel \((\mathcal{X}, p(y|x), \mathcal{Y})\) is:

1. An index set \(\{1, 2, \ldots, M\}\)

2. An encoding function \(X^n : \{1, 2, \ldots, M\} \rightarrow \mathcal{X}^n\) yielding codewords \(X^n(1), X^n(2), X^n(3), \ldots, X^n(M)\). Each source message has a codeword, and each codeword is \(n\) code symbols.

3. Decoding function, i.e., \(g : \mathcal{Y}^n \rightarrow \{1, 2, \ldots, M\}\) which makes a “guess” about original message given channel output.

- In an \((M, n)\) code, \(M = \) the number of possible messages to be sent, and \(n = \) number of channel uses by the codewords of the code.
Definition 3.1 (Probability of Error $\lambda_i$ for message $i \in \{1, \ldots, M\}$)

$$\lambda_i \triangleq \Pr(g(Y^n) \neq i | X^n = X^n(i)) = \sum_{y^n \in \mathcal{Y}^n} p(y^n | X^n(i)) 1(g(y^n) \neq i)$$ (11)
Definition 3.1 (Probability of Error $\lambda_i$ for message $i \in \{1, \ldots, M\}$)

$$\lambda_i \triangleq \Pr(g(Y^n) \neq i | X^n = X^n(i)) = \sum_{y^n \in Y^n} p(y^n | X^n(i)) 1(g(y^n) \neq i)$$  \hspace{1cm} (11)

Definition 3.2 (Max probability of Error $\lambda^{(n)}$ for $(M, n)$ code)

$$\lambda^{(n)} \triangleq \max_{i \in \{1,2,\ldots,M\}} \lambda_i$$  \hspace{1cm} (12)
Definition 3.3 (Average probability of error $P_e^{(n)}$)

$$P_e^{(n)} = \frac{1}{M} \sum_{i=1}^{M} \lambda_i = \Pr(I \neq g(Y^n))$$

(13)

where $I$ is a r.v. with probability $\Pr(I = i)$ according to a uniform source distribution . . .

$$= E(1(I \neq g(Y^n))) = \sum_{i=1}^{M} \Pr(g(Y^n) \neq i|X^n = X^n(i))p(i)$$

(14)

with $p(i) = 1/M$. 

A key Shannon’s result is that a small average probability of error means we must have a small maximum probability of error!
Definition 3.3 (Average probability of error $P_{e}^{(n)}$)

$$P_{e}^{(n)} = \frac{1}{M} \sum_{i=1}^{M} \lambda_i = \Pr(I \neq g(Y^n))$$

(13)

where $I$ is a r.v. with probability $\Pr(I = i)$ according to a uniform source distribution . . .

$$= E(1(I \neq g(Y^n))) = \sum_{i=1}^{M} \Pr(g(Y^n) \neq i | X^n = X^n(i))p(i)$$

(14)

with $p(i) = 1/M$.

A key Shannon’s result is that a small average probability of error means we must have a small maximum probability of error!
Definition 3.4 (Rate $R$ of an $(M, n)$ code)

$$R = \frac{\log M}{n} = \frac{\text{total num. of bits in a source message}}{\text{total num. of channel uses needed to send a message}}$$

(15)
Definition 3.4 (Rate $R$ of an $(M, n)$ code)

$$R = \frac{\log M}{n} = \frac{\text{total num. of bits in a source message}}{\text{total num. of channel uses needed to send a message}} \quad (15)$$

- The rate $R$ is in units of bits per channel use, or bits per transmission.
Rate

Definition 3.4 (Rate $R$ of an $(M, n)$ code)

$R = \frac{\log M}{n} = \frac{\text{total num. of bits in a source message}}{\text{total num. of channel uses needed to send a message}}$

(15)

- The rate $R$ is in units of bits per channel use, or bits per transmission.

Definition 3.5 (Achievability for a given channel)

A given rate $R$ is achievable for a given channel if $\exists$ a sequence of $(\lceil 2^{nR} \rceil, n)$ codes such that the maximal probability of error $\lambda^{(n)} \to 0$ as $n \to \infty$. 

Definition 3.6 (Capacity of a DMC)

The capacity of a DMC is the largest possible achievable rate.

- So the capacity of a DMC is the rate beyond which the error won’t any longer go to zero with increasing $n$. 
Definition 3.6 (Capacity of a DMC)

The capacity of a DMC is the largest possible achievable rate.

- So the capacity of a DMC is the rate beyond which the error won’t any longer go to zero with increasing $n$.
- Note: this is a different notion of capacity that we encountered before.
**Definition 3.6 (Capacity of a DMC)**

The **capacity** of a DMC is the largest possible achievable rate.

- So the capacity of a DMC is the rate beyond which the error won’t any longer go to zero with increasing $n$.
- Note: this is a different notion of capacity that we encountered before.
- Before we defined $C = \max_{p(x)} I(X; Y)$. 
Definition 3.6 (Capacity of a DMC)

The capacity of a DMC is the largest possible achievable rate.

- So the capacity of a DMC is the rate beyond which the error won’t any longer go to zero with increasing $n$.
- Note: this is a different notion of capacity that we encountered before.
- Before we defined $C = \max_{p(x)} I(X; Y)$.
- Here we are defining something called the “capacity of a DMC”.
Definition 3.6 (Capacity of a DMC)

The capacity of a DMC is the largest possible achievable rate.

- So the capacity of a DMC is the rate beyond which the error won’t any longer go to zero with increasing $n$.
- Note: this is a different notion of capacity that we encountered before.
- Before we defined $C = \max_p p(x) \cdot I(X; Y)$.
- Here we are defining something called the “capacity of a DMC”.
- We have not yet compared the two
Definition 3.6 (Capacity of a DMC)

The capacity of a DMC is the largest possible achievable rate.

- So the capacity of a DMC is the rate beyond which the error won’t any longer go to zero with increasing $n$.
- Note: this is a different notion of capacity that we encountered before.
- Before we defined $C = \max_{p(x)} I(X; Y)$.
- Here we are defining something called the “capacity of a DMC”.
- We have not yet compared the two (but of course we will 😊).
Joint Typicality

Definition 3.7 (Joint typicality of a set of sequences)

A set of sequences \( \{(x_1:n, y_1:n)\} \) w.r.t. \( p(x, y) \) is jointly typical (\( \in A^{(n)}_\epsilon \)) as per the following definition:

\[
A^{(n)}_\epsilon = \left\{ (x^n, y^n) \in \mathcal{X}^n \times \mathcal{Y}^n : \right. \\
\left. \left| \frac{1}{n} \log p(x^n) - H(X) \right| < \epsilon, \text{ } x \text{-typical} \right. \\
\left. \left| \frac{1}{n} \log p(y^n) - H(Y) \right| < \epsilon, \text{ } y \text{-typical} \right. \\
\left. \left| \frac{1}{n} \log p(x^n, y^n) - H(X,Y) \right| < \epsilon, \text{ } (x,y) \text{-typical} \right\}
\]  

(16)

with \( p(x^n, y^n) = \prod_{i=1}^{n} p(x_i, y_i) \).
Definition 3.7 (Joint typicality of a set of sequences)

A set of sequences $\{(x_{1:n}, y_{1:n})\}$ w.r.t. $p(x, y)$ is jointly typical ($\in A^{(n)}_\epsilon$) as per the following definition:

$$A^{(n)}_\epsilon = \left\{(x^n, y^n) \in \mathcal{X}^n \times \mathcal{Y}^n : \right. \right.$$  \[ a) \left. \left| - \frac{1}{n} \log p(x^n) - H(X) \right| < \epsilon, \quad x\text{-typical} \right. \]  \[ b) \quad \left| - \frac{1}{n} \log p(y^n) - H(Y) \right| < \epsilon, \quad y\text{-typical} \right. \]  \[ c) \left. \left| - \frac{1}{n} \log p(x^n, y^n) - H(X,Y) \right| < \epsilon, \quad (x, y)\text{-typical} \right. \]  \[ \left. \right\} \]  \[ \text{with } p(x^n, y^n) = \prod_{i=1}^{n} p(x_i, y_i). \]
Definition 3.7 (Joint typicality of a set of sequences)

A set of sequences \( \{(x_{1:n}, y_{1:n})\} \) w.r.t. \( p(x, y) \) is jointly typical \((\in A^{(n)}_\epsilon)\) as per the following definition:

\[
A^{(n)}_\epsilon = \left\{ (x^n, y^n) \in \mathcal{X}^n \times \mathcal{Y}^n : \right. \\
\begin{align}
a) & \quad \left| -\frac{1}{n} \log p(x^n) - H(X) \right| < \epsilon, \quad x\text{-typical} \\
b) & \quad \left| -\frac{1}{n} \log p(y^n) - H(Y) \right| < \epsilon, \quad y\text{-typical} \\
\end{align}
\]

\[
\left\} \right. \\
\]

with \( p(x^n, y^n) = \prod_{i=1}^{n} p(x_i, y_i) \).
Definition 3.7 (Joint typicality of a set of sequences)

A set of sequences \( \{(x_{1:n}, y_{1:n})\} \) w.r.t. \( p(x, y) \) is jointly typical \( (\in A_{\epsilon}^{(n)}) \) as per the following definition:

\[
A_{\epsilon}^{(n)} = \left\{ (x^n, y^n) \in \mathcal{X}^n \times \mathcal{Y}^n : \right. \\
\left. a) \quad \left| -\frac{1}{n} \log p(x^n) - H(X) \right| < \epsilon, \quad x\text{-typical} \right. \\
\left. b) \quad \left| -\frac{1}{n} \log p(y^n) - H(Y) \right| < \epsilon, \quad y\text{-typical} \right. \\
\left. \quad \text{and} \quad c) \quad \left| -\frac{1}{n} \log p(x^n, y^n) - H(X, Y) \right| < \epsilon, \quad (x, y)\text{-typical} \right. \\
\left. \right\} 
\]

with \( p(x^n, y^n) = \prod_{i=1}^{n} p(x_i, y_i) \).
Jointly Typical Sequences: Picture

set of all jointly typical pairs of sequences \( 2^{nH(X,Y)} \)

set of all pairs of marginally typical sequences

\( 2^{nH(X)} \)

\( 2^{nH(Y)} \)

\( 2^{nH(Y|X)} \)

\( 2^{nH(X|Y)} \)
So intuitively,

\[
\frac{\text{num. jointly typical seqs.}}{\text{num ind. chosen typical seqs.}} = \frac{2^n H(X,Y)}{2^n H(X) 2^n H(Y)} = 2^{n(H(X,Y) - H(X) - H(Y))} = 2^{-n I(X;Y)}
\]
Jointly Typical Sequences: Intuition

- So intuitively,

\[
\frac{\text{num. jointly typical seqs.}}{\text{num ind. chosen typical seqs.}} = \frac{2^{nH(X,Y)}}{2^{nH(X)}2^{nH(Y)}}
\]

(20)

\[
= 2^{n(H(X,Y) - H(X) - H(Y))}
\]

(21)

\[
= 2^{-nI(X;Y)}
\]

(22)

- So if we independently at random choose two (singly) typical sequences for \(X\) and \(Y\), then the chance that it will be an \((X, Y)\) jointly typical sequence decreases exponentially with \(n\), as long as \(I(X; Y) > 0\).
Jointly Typical Sequences: Intuition

- So intuitively,

\[
\frac{\text{num. jointly typical seqs.}}{\text{num ind. chosen typical seqs.}} = \frac{2^{nH(X,Y)}}{2^{nH(X)}2^{nH(Y)}} = 2^{n(H(X,Y)-H(X)-H(Y))} = 2^{-nI(X;Y)}
\]  

(20)

(21)

(22)

- So if we independently at random choose two (singly) typical sequences for \(X\) and \(Y\), then the chance that it will be an \((X,Y)\) jointly typical sequence decreases exponentially with \(n\), as long as \(I(X;Y) > 0\).

- to decrease this chance as much as possible, it can become \(2^{-nC}\).
Joint AEP

Theorem 4.1

Let \((X^n, Y^n) \sim p(x^n, y^n) = \prod_{i=1}^{n} p(x_i, y_i)\). Then

\[
\Pr((X^n, Y^n) \in A(\epsilon^n)) \rightarrow 1 \quad \text{as} \quad n \rightarrow \infty.
\]

\[
|A(\epsilon^n)| \leq 2^n (H(X,Y) + \epsilon) \quad \text{and} \quad (1 - \epsilon)^2 n (H(X,Y) - \epsilon) \leq |A(\epsilon^n)|.
\]

If \((\tilde{X}^n, \tilde{Y}^n) \sim p(x^n) p(y^n)\) are drawn independently, then

\[
\Pr((\tilde{X}^n, \tilde{Y}^n) \in A(\epsilon^n)) \leq 2 - n(I(X;Y) - 3\epsilon) \quad \text{(23)}
\]

and for sufficiently large \(n\), we have

\[
\Pr((\tilde{X}^n, \tilde{Y}^n) \in A(\epsilon^n)) \geq (1 - \epsilon)^2 - n(I(X;Y) + 3\epsilon) \quad \text{(24)}
\]

Key property: we have bound on the probability of independently drawn sequences being jointly typical, falls off exponentially fast with \(n\), if \(I(X;Y) > 0\).
Joint AEP

Theorem 4.1

Let \((X^n, Y^n) \sim p(x^n, y^n) = \prod_{i=1}^{n} p(x_i, y_i)\). Then

1. \(Pr \left( (X^n, Y^n) \in A^{(n)}_c \right) \to 1 \text{ as } n \to \infty.\)
Joint AEP

Theorem 4.1

Let \((X^n, Y^n) \sim p(x^n, y^n) = \prod_{i=1}^{n} p(x_i, y_i)\). Then

1. \(\Pr \left( (X^n, Y^n) \in A_{\epsilon}^{(n)} \right) \rightarrow 1 \text{ as } n \rightarrow \infty.\)

2. \(|A_{\epsilon}^{(n)}| \leq 2^{n(H(X,Y)+\epsilon)}\) and \((1 - \epsilon)2^{n(H(X,Y)-\epsilon)} \leq |A_{\epsilon}^{(n)}|\).
Joint AEP

Theorem 4.1

Let \( (X^n, Y^n) \sim p(x^n, y^n) = \prod_{i=1}^{n} p(x_i, y_i) \). Then

1. \( \Pr \left( (X^n, Y^n) \in A^{(n)}_\epsilon \right) \to 1 \) as \( n \to \infty \).

2. \( |A^{(n)}_\epsilon| \leq 2^n(H(X,Y)+\epsilon) \) and \( (1-\epsilon)2^n(H(X,Y)-\epsilon) \leq |A^{(n)}_\epsilon| \).

3. If \( (\tilde{X}^n, \tilde{Y}^n) \sim p(x^n)p(y^n) \) are drawn independently, then

\[ \Pr \left( (\tilde{X}^n, \tilde{Y}^n) \in A^{(n)}_\epsilon \right) \leq 2^{-n(I(X;Y)-3\epsilon)} \] (23)

and for sufficiently large \( n \), we have

\[ \Pr \left( (\tilde{X}^n, \tilde{Y}^n) \in A^{(n)}_\epsilon \right) \geq (1-\epsilon)2^{-n(I(X;Y)+3\epsilon)} \] (24)
Theorem 4.1

Let $(X^n, Y^n) \sim p(x^n, y^n) = \prod_{i=1}^{n} p(x_i, y_i)$. Then

1. $\Pr \left( (X^n, Y^n) \in A_{\epsilon}^{(n)} \right) \to 1$ as $n \to \infty$.

2. $|A_{\epsilon}^{(n)}| \leq 2^{n(H(X,Y)+\epsilon)}$ and $(1 - \epsilon)2^{n(H(X,Y)-\epsilon)} \leq |A_{\epsilon}^{(n)}|$.  

3. If $(\tilde{X}^n, \tilde{Y}^n) \sim p(x^n)p(y^n)$ are drawn independently, then

   \[ \Pr \left( (\tilde{X}^n, \tilde{Y}^n) \in A_{\epsilon}^{(n)} \right) \leq 2^{-n(I(X;Y)-3\epsilon)} \tag{23} \]

   and for sufficiently large $n$, we have

   \[ \Pr \left( (\tilde{X}^n, \tilde{Y}^n) \in A_{\epsilon}^{(n)} \right) \geq (1 - \epsilon)2^{-n(I(X;Y)+3\epsilon)} \tag{24} \]

- Key property: we have bound on the probability of independently drawn sequences being jointly typical, falls off exponentially fast with $n$, if $I(X;Y) > 0$. 

Prof. Jeff Bilmes
Joint AEP proof

Proof of $\Pr \left( (X^n, Y^n) \in A_\epsilon^{(n)} \right) \to 1$. 

...
Joint AEP proof

Proof of \( \Pr \left( (X^n, Y^n) \in A^{(n)}_\epsilon \right) \to 1. \)

1. We have, by the w.l.l.n.s,

\[
-\frac{1}{n} \log \Pr(X^n) \to -E(\log p(X)) = H(X) \quad (25)
\]
Joint AEP proof

Proof of $\Pr\left( (X^n, Y^n) \in A^{(n)}_\epsilon \right) \to 1$.

- We have, by the w.l.l.n.s,

$$\frac{1}{n} \log \Pr(X^n) \to -E(\log p(X)) = H(X)$$ \hspace{1cm} (25)

so $\forall \epsilon > 0$, $\exists m_1$ such that for $n > m_1$

$$\Pr\left( \left| \frac{1}{n} \log \Pr(X^n) - H(X) \right| > \epsilon \right) < \epsilon/3$$ \hspace{1cm} (26)

$\Pr$ call this $S_1$
Proof of $\Pr \left( (X^n, Y^n) \in A^{(n)}_{\epsilon} \right) \rightarrow 1$.

- We have, by the w.l.l.n.s,

$$\frac{1}{n} \log \Pr(X^n) \rightarrow -E(\log p(X)) = H(X)$$

so $\forall \epsilon > 0$, $\exists m_1$ such that for $n > m_1$

$$\Pr \left( \left| \frac{1}{n} \log \Pr(X^n) - H(X) \right| > \epsilon \right) < \epsilon/3$$

call this $S_1$

- So, $S_1$ is a non-typical event.

...
Joint AEP proof

Proof of $\Pr \left( (X^n, Y^n) \in A_e^{(n)} \right) \to 1$. 

...
Joint AEP proof

Proof of $\Pr \left( (X^n, Y^n) \in A_{\epsilon}^{(n)} \right) \to 1$.

- Also, $\exists m_2, m_3$ such that $\forall n > m_2$, we have

$$\Pr \left( \left| -\frac{1}{n} \log \Pr(Y^n) - H(Y) \right| > \epsilon \right) < \epsilon/3 \quad (27)$$

...
Joint AEP proof

Proof of $\Pr \left( (X^n, Y^n) \in A^{(n)}_\epsilon \right) \to 1$.

- Also, $\exists m_2, m_3$ such that $\forall n > m_2$, we have

$$\Pr \left( \left| \frac{1}{n} \log \Pr(Y^n) - H(Y) \right| > \epsilon \right) < \epsilon/3$$

(call this $S_2$)  \hspace{1cm} (27)

and $\forall n > m_3$, we have

$$\Pr \left( \left| \frac{1}{n} \log \Pr(X^n, Y^n) - H(X,Y) \right| > \epsilon \right) < \epsilon/3$$

(call this $S_3$)  \hspace{1cm} (28)

...
Joint AEP proof

Proof of $\Pr\left( (X^n, Y^n) \in A_{\epsilon}^{(n)} \right) \to 1$.

- Also, $\exists m_2, m_3$ such that $\forall n > m_2$, we have

  \[
  \Pr \left( \left| - \frac{1}{n} \log \Pr(Y^n) - H(Y) \right| > \epsilon \right) < \epsilon/3 \tag{27}
  \]

  call this $S_2$

  and $\forall n > m_3$, we have

  \[
  \Pr \left( \left| - \frac{1}{n} \log \Pr(X^n, Y^n) - H(X, Y) \right| > \epsilon \right) < \epsilon/3 \tag{28}
  \]

  call this $S_3$

- So all events $S_1$, $S_2$ and $S_3$ are non-typical events.
Joint AEP proof

Proof of $\Pr \left( (X^n, Y^n) \in A_{\epsilon}^{(n)} \right) \rightarrow 1.$
Joint AEP proof

Proof of $\Pr \left( (X^n, Y^n) \in A^{(n)}_\epsilon \right) \rightarrow 1$.

- For $n > \max(m_1, m_2, m_3)$, we have that $p(S_1 \cup S_2 \cup S_3) \leq \epsilon = 3\epsilon/3$ by the union bound.
Joint AEP proof

Proof of $\Pr \left( (X^n, Y^n) \in A_{\epsilon}^{(n)} \right) \to 1$.

- For $n > \max(m_1, m_2, m_3)$, we have that $p(S_1 \cup S_2 \cup S_3) \leq \epsilon = 3\epsilon/3$ by the union bound.

- So, non-typicality has probability $< \epsilon$, meaning $\Pr(A_{\epsilon}^{(n)c}) \leq \epsilon$ giving $\Pr(A_{\epsilon}^{(n)}) \geq 1 - \epsilon$, as desired. □ for 1.
Proof of $|A_\epsilon^{(n)}| \leq 2^n(H(X,Y)+\epsilon)$. 
Proof of \( |A^{(n)}_\epsilon| \leq 2^n(H(X,Y)+\epsilon) \).

We have

\[
1 = \sum_{x^n,y^n} p(x^n, y^n) \geq \sum_{(x^n,y^n) \in A^{(n)}_\epsilon} p(x^n, y^n) \geq |A^{(n)}_\epsilon| 2^{-n(H(X,Y)+\epsilon)}
\]

\[\Rightarrow |A^{(n)}_\epsilon| \leq 2^n(H(X,Y)+\epsilon)\]  

(29)  

(30)
Joint AEP proof

Proof of $|A_{\epsilon}^{(n)}| \leq 2^n(H(X,Y)+\epsilon)$.

- We have

$$1 = \sum_{x^n,y^n} p(x^n, y^n) \geq \sum_{(x^n,y^n)\in A_{\epsilon}^{(n)}} p(x^n, y^n) \geq |A_{\epsilon}^{(n)}|2^{-n(H(X,Y)+\epsilon)}$$

$$\Rightarrow |A_{\epsilon}^{(n)}| \leq 2^n(H(X,Y)+\epsilon) \quad (29)$$

- Also, from before, $\Pr(A_{\epsilon}^{(n)}) \geq 1 - \epsilon$ for big $n$, giving:

$$1 - \epsilon \leq \sum_{(x^n,y^n)\in A_{\epsilon}^{(n)}} p(x^n, y^n) \leq |A_{\epsilon}^{(n)}|2^{-n(H(X,Y)-\epsilon)}$$

$$\Rightarrow |A_{\epsilon}^{(n)}| \geq (1 - \epsilon)2^n(H(X,Y)-\epsilon) \quad (31)$$

□ for 2.
Proof of two indep. sequences are likely not jointly typical.
Proof of two indep. sequences are likely not jointly typical.

- Let $\tilde{X}^n, \tilde{Y}^n$ be independent $\sim p(x^n)p(y^n)$, i.e. the two sequences are independent of each other.
Joint AEP proof

Proof of two indep. sequences are likely not jointly typical.

- Let $\tilde{X}^n, \tilde{Y}^n$ be independent $\sim p(x^n)p(y^n)$, i.e. the two sequences are independent of each other.

- Then we have the following two derivations:

\[
\Pr \left( (\tilde{X}^n, \tilde{Y}^n) \in A_\epsilon^{(n)} \right) = \sum_{(x^n, y^n) \in A_\epsilon^{(n)}} p(x^n)p(y^n) \tag{33}
\]

\[
\leq 2^{n(H(X,Y)+\epsilon)} 2^{-n(H(X)-\epsilon)} 2^{-n(H(Y)-\epsilon)} \tag{34}
\]

\[
= 2^{-n(I(X;Y)-3\epsilon)} \tag{35}
\]
Joint AEP proof

Proof of two indep. sequences are likely not jointly typical.

1. Let $\tilde{X}^n, \tilde{Y}^n$ be independent $\sim p(x^n)p(y^n)$, i.e. the two sequences are independent of each other.

2. Then we have the following two derivations:

\[
\Pr \left( (\tilde{X}^n, \tilde{Y}^n) \in A_{\epsilon}^{(n)} \right) = \sum_{(x^n,y^n) \in A_{\epsilon}^{(n)}} p(x^n)p(y^n) \tag{33}
\]
\[
\leq 2^{n(H(X,Y)+\epsilon)}2^{-n(H(X)-\epsilon)}2^{-n(H(Y)-\epsilon)} \tag{34}
\]
\[
= 2^{-n(I(X;Y)-3\epsilon)} \tag{35}
\]

\[
\Pr \left( (\tilde{X}^n, \tilde{Y}^n) \in A_{\epsilon}^{(n)} \right) \geq (1 - \epsilon)2^{n(H(X,Y)-\epsilon)}2^{-n(H(X)+\epsilon)}2^{-n(H(Y)+\epsilon)} \tag{36}
\]
\[
= (1 - \epsilon)2^{-n(I(X;Y)+3\epsilon)}
\]
Another Intuitive (and somewhat redundant) Reprieve

- There are $\approx 2^{nH(X)}$ typical $X$ sequences
Another Intuitive (and somewhat redundant) Reprieve

- There are \( \approx 2^{nH(X)} \) typical \( X \) sequences
- There are \( \approx 2^{nH(Y)} \) typical \( Y \) sequences.
Another Intuitive (and somewhat redundant) Reprieve

- There are $\approx 2^{nH(X)}$ typical $X$ sequences
- There are $\approx 2^{nH(Y)}$ typical $Y$ sequences.
- The total number of independent typical pairs is $\approx 2^{nH(X)}2^{nH(Y)}$
Another Intuitive (and somewhat redundant) Reprieve

- There are $\approx 2^{nH(X)}$ typical $X$ sequences.
- There are $\approx 2^{nH(Y)}$ typical $Y$ sequences.
- The total number of independent typical pairs is $\approx 2^{nH(X)}2^{nH(Y)}$, but not all of them are jointly typical. Rather only $\approx 2^{nH(X,Y)}$ of them are jointly typical.
Another Intuitive (and somewhat redundant) Reprieve

- There are $\approx 2^{nH(X)}$ typical $X$ sequences.
- There are $\approx 2^{nH(Y)}$ typical $Y$ sequences.
- The total number of independent typical pairs is $\approx 2^{nH(X)}2^{nH(Y)}$, but not all of them are jointly typical. Rather only $\approx 2^{nH(X,Y)}$ of them are jointly typical.
- The fraction of independent typical sequences that are jointly typical is:

$$\frac{2^{nH(X,Y)}}{2^{nH(X)}2^{nH(Y)}} = 2^{n(H(X,Y) - H(X) - H(Y))} = 2^{-nI(X,Y)}$$

(37)
There are $\approx 2^{nH(X)}$ typical $X$ sequences.

There are $\approx 2^{nH(Y)}$ typical $Y$ sequences.

The total number of independent typical pairs is $\approx 2^{nH(X)}2^{nH(Y)}$, but not all of them are jointly typical. Rather only $\approx 2^{nH(X,Y)}$ of them are jointly typical.

The fraction of independent typical sequences that are jointly typical is:

$$\frac{2^{nH(X,Y)}}{2^{nH(X)}2^{nH(Y)}} = 2^n(H(X,Y) - H(X) - H(Y)) = 2^{-nI(X,Y)}$$

(37)

and this is essentially the probability that a randomly chosen pair of (marginally) typical sequences is jointly typical.
Jointly Typical Sequences: Picture

- Set of all jointly typical pairs of sequences: $2^{nH(X,Y)}$
- Set of all pairs of marginally typical sequences:
  - $2^{nH(X)}$
  - $2^{nH(Y)}$
  - $2^{nH(X|Y)}$
  - $2^{nH(Y|X)}$
More Intuition

- So if we use typicality to decode (which we will) then there are about $2^{nI(X;Y)}$ pairs of sequences available before we start needing to use pairs that would be jointly typical if chosen randomly.
More Intuition

- So if we use typicality to decode (which we will) then there are about $2^{nI(X;Y)}$ pairs of sequences available before we start needing to use pairs that would be jointly typical if chosen randomly.

- Ex: if $p(x) = 1/M$ then we can choose about $M$ samples before we see a given particular $x$, on average.
The basic idea is to use joint typicality.
The basic idea is to use joint typicality.

Given a received codeword $y^n$, find an $x^n$ that is jointly typical with $y^n$. 
The basic idea is to use joint typicality.

Given a received codeword $y^n$, find an $x^n$ that is jointly typical with $y^n$.

This $x^n$ will occur jointly with $y^n$ with probability 1, for large enough $n$. 
Channel Coding Theorem (Shannon 1948)

- The basic idea is to use joint typicality.
- Given a received codeword $y^n$, find an $x^n$ that is jointly typical with $y^n$.
- This $x^n$ will occur jointly with $y^n$ with probability 1, for large enough $n$.
- Also, the probability that some other $\hat{x}^n$ is jointly typical with $y^n$ is about $2^{-nI(X;Y)}$. 
The basic idea is to use joint typicality.

Given a received codeword $y^n$, find an $x^n$ that is jointly typical with $y^n$.

This $x^n$ will occur jointly with $y^n$ with probability 1, for large enough $n$.

Also, the probability that some other $\hat{x}^n$ is jointly typical with $y^n$ is about $2^{-nI(X;Y)}$,

so if we use $<2^{nI(X;Y)}$ codewords, then some other sequence being jointly typical will occur with vanishingly small probability for large $n$. 
Theorem 5.1

All rates below $C$ are achievable. Specifically, $\forall R < C$, there exists a sequence of $(2^n R, n)$ codes with maximum probability of error $\lambda(n) \rightarrow 0$ as $n \rightarrow \infty$.

Conversely, any $(2^n R, n)$ sequence of codes with $\lambda(n) \rightarrow 0$ as $n \rightarrow \infty$ must have that $R < C$.

Implications: as long as we do not code above capacity we can, for all intents and purposes, code with zero error. This is true for all noisy channels representable under this model. We're talking about discrete channels now, but we generalize this to continuous channels in the coming weeks.
Theorem 5.1

All rates below \( C \) are achievable. Specifically, \( \forall R < C \), there exists a sequence of \( (2^{nR}, n) \) codes with maximum probability of error \( \lambda^{(n)} \rightarrow 0 \) as \( n \rightarrow \infty \).
Channel Coding Theorem (Shannon 1948): more formally

Theorem 5.1

All rates below $C$ are achievable. Specifically, $\forall R < C$, there exists a sequence of $(2^{nR}, n)$ codes with maximum probability of error $\lambda^{(n)} \to 0$ as $n \to \infty$. Conversely, any $(2^{nR}, n)$ sequence of codes with $\lambda^{(n)} \to 0$ as $n \to \infty$ must have that $R < C$. 
Channel Coding Theorem (Shannon 1948): more formally

**Theorem 5.1**

All rates below $C$ are achievable. Specifically, $\forall R < C$, there exists a sequence of $(2^{nR}, n)$ codes with maximum probability of error $\lambda(n) \to 0$ as $n \to \infty$. Conversely, any $(2^{nR}, n)$ sequence of codes with $\lambda(n) \to 0$ as $n \to \infty$ must have that $R < C$.

- Implications: as long as we do not code above capacity we can, for all intents and purposes, code with zero error.
Channel Coding Theorem (Shannon 1948): more formally

Theorem 5.1

All rates below $C$ are achievable. Specifically, $\forall R < C$, there exists a sequence of $(2^{nR}, n)$ codes with maximum probability of error $\lambda^{(n)} \to 0$ as $n \to \infty$. Conversely, any $(2^{nR}, n)$ sequence of codes with $\lambda^{(n)} \to 0$ as $n \to \infty$ must have that $R < C$.

- Implications: as long as we do not code above capacity we can, for all intents and purposes, code with zero error.
- This is true for all noisy channels representable under this model.
Channel Coding Theorem (Shannon 1948): more formally

**Theorem 5.1**

All rates below $C$ are achievable. Specifically, $\forall R < C$, there exists a sequence of $(2^{nR}, n)$ codes with maximum probability of error $\lambda^{(n)} \to 0$ as $n \to \infty$. Conversely, any $(2^{nR}, n)$ sequence of codes with $\lambda^{(n)} \to 0$ as $n \to \infty$ must have that $R < C$.

- Implications: as long as we do not code above capacity we can, for all intents and purposes, code with zero error.
- This is true for all noisy channels representable under this model.
- We’re talking about discrete channels now, but we generalize this to continuous channels in the coming weeks.
**Channel Theorem**

- We could look at error for a particular code and bound its errors.
Channel Theorem

- We could look at error for a particular code and bound its errors.
- Instead, we look at the average probability of errors of all codes generated randomly.
Channel Theorem

- We could look at error for a particular code and bound its errors.
- Instead, we look at the average probability of errors of all codes generated randomly.
- We then prove that this average error is small.
Channel Theorem

- We could look at error for a particular code and bound its errors.
- Instead, we look at the average probability of errors of all codes generated randomly.
- We then prove that this average error is small.
- This implies $\exists$ many good codes to make the average small.
Channel Theorem

- We could look at error for a particular code and bound its errors.
- Instead, we look at the average probability of errors of all codes generated randomly.
- We then prove that this average error is small.
- This implies \( \exists \) many good codes to make the average small.
- To show that the maximum probability of error also small, we throw away the worst 50% of the codes.
Channel Theorem

- We could look at error for a particular code and bound its errors.
- Instead, we look at the average probability of errors of all codes generated randomly.
- We then prove that this average error is small.
- This implies \( \exists \) many good codes to make the average small.
- To show that the maximum probability of error also small, we throw away the worst 50% of the codes.
- Recall: idea is, for a given channel \((\mathcal{X}, p(y|x), \mathcal{Y})\) come up with a \((2^{nR}, n)\) code of rate \(R\) which means we need:
Channel Theorem

- We could look at error for a particular code and bound its errors.
- Instead, we look at the average probability of errors of all codes generated randomly.
- We then prove that this average error is small.
- This implies \( \exists \) many good codes to make the average small.
- To show that the maximum probability of error also small, we throw away the worst 50% of the codes.
- Recall: idea is, for a given channel \((X, p(y|x), Y)\) come up with a \((2^{nR}, n)\) code of rate \(R\) which means we need:
  1. Index set \(\{1, \ldots, M\}\)
Channel Theorem

- We could look at error for a particular code and bound its errors.
- Instead, we look at the average probability of errors of all codes generated randomly.
- We then prove that this average error is small.
- This implies ∃ many good codes to make the average small.
- To show that the maximum probability of error also small, we throw away the worst 50% of the codes.
- Recall: idea is, for a given channel \((X, p(y|x), Y)\) come up with a \((2^{nR}, n)\) code of rate \(R\) which means we need:
  1. Index set \(\{1, \ldots, M\}\)
  2. Encoder: \(X^n : \{1, \ldots, M\} \rightarrow X^n\) maps to codewords \(X^n(i)\)
Channel Theorem

- We could look at error for a particular code and bound its errors.
- Instead, we look at the average probability of errors of all codes generated randomly.
- We then prove that this average error is small.
- This implies \( \exists \) many good codes to make the average small.
- To show that the maximum probability of error also small, we throw away the worst 50% of the codes.

Recall: idea is, for a given channel \((\mathcal{X}, p(y|x), \mathcal{Y})\) come up with a \((2^{nR}, n)\) code of rate \(R\) which means we need:

1. Index set \(\{1, \ldots, M\}\)
2. Encoder: \(X^n : \{1, \ldots, M\} \to \mathcal{X}^n\) maps to codewords \(X^n(i)\)
3. Decoder: \(g : \mathcal{Y}^n \to \{1, \ldots, M\}\).
Channel Theorem

- We could look at error for a particular code and bound its errors.
- Instead, we look at the average probability of errors of all codes generated randomly.
- We then prove that this average error is small.
- This implies ∃ many good codes to make the average small.
- To show that the maximum probability of error also small, we throw away the worst 50% of the codes.
- Recall: idea is, for a given channel \((\mathcal{X}, p(y|x), \mathcal{Y})\) come up with a \((2^{nR}, n)\) code of rate \(R\) which means we need:
  1. Index set \(\{1, \ldots, M\}\)
  2. Encoder: \(X^n : \{1, \ldots, M\} \rightarrow \mathcal{X}^n\) maps to codewords \(X^n(i)\)
  3. Decoder: \(g : \mathcal{Y}^n \rightarrow \{1, \ldots, M\}\).
- Two parts to prove: 1) all rates \(R < C\) are achievable (exists a code with vanishing error). Conversely, 2) if the error goes to zero, then must have \(R < C\).
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

- Given $R < C$, assume use of $p(x)$ and generate $2^{nR}$ random codewords using $p(x^n) = \prod_{i=1}^{n} p(x_i)$.
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

- Given $R < C$, assume use of $p(x)$ and generate $2^{nR}$ random codewords using $p(x^n) = \prod_{i=1}^{n} p(x_i)$.
- Choose $p(x)$ arbitrarily for now, and then change it later to get $C$.
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

- Given $R < C$, assume use of $p(x)$ and generate $2^{nR}$ random codewords using $p(x^n) = \prod_{i=1}^{n} p(x_i)$.
- Choose $p(x)$ arbitrarily for now, and then change it later to get $C$.
- Set of random codewords (the codebook) can be seen as a matrix:

$$
C = \begin{bmatrix}
x_1(1) & x_2(1) & x_3(1) & \ldots & x_n(1) \\
x_1(2) & x_2(2) & x_3(2) & \ldots & x_n(2) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\
x_1(2^{nR}) & x_2(2^{nR}) & x_3(2^{nR}) & \ldots & x_n(2^{nR})
\end{bmatrix}
$$

(38)
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

- Given $R < C$, assume use of $p(x)$ and generate $2^{nR}$ random codewords using $p(x^n) = \prod_{i=1}^{n} p(x_i)$.
- Choose $p(x)$ arbitrarily for now, and then change it later to get $C$.
- Set of random codewords (the codebook) can be seen as a matrix:

$$
C = \begin{bmatrix}
    x_1(1) & x_2(1) & x_3(1) & \ldots & x_n(1) \\
    x_1(2) & x_2(2) & x_3(2) & \ldots & x_n(2) \\
    \vdots & \vdots & \vdots & \ddots & \vdots \\
    x_1(2^{nR}) & x_2(2^{nR}) & x_3(2^{nR}) & \ldots & x_n(2^{nR})
\end{bmatrix} \quad (38)
$$

- So, there are $2^{nR}$ codes each of length $n$ generated via $p(x)$. 

...
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

- Given $R < C$, assume use of $p(x)$ and generate $2^{nR}$ random codewords using $p(x^n) = \prod_{i=1}^{n} p(x_i)$.
- Choose $p(x)$ arbitrarily for now, and then change it later to get $C$.
- Set of random codewords (the codebook) can be seen as a matrix:

$$C = \begin{bmatrix}
    x_1(1) & x_2(1) & x_3(1) & \ldots & x_n(1) \\
    x_1(2) & x_2(2) & x_3(2) & \ldots & x_n(2) \\
    \vdots & \vdots & \vdots & \ddots & \vdots \\
    x_1(2^{nR}) & x_2(2^{nR}) & x_3(2^{nR}) & \ldots & x_n(2^{nR})
\end{bmatrix} \tag{38}$$

- So, there are $2^{nR}$ codes each of length $n$ generated via $p(x)$.
- To send any message $\omega \in \{1, 2, \ldots, M = 2^{nR}\}$, we send codeword $x_{1:n}(\omega) = (x_1(\omega), x_2(\omega), \ldots, x_n(\omega))$. 

...
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

- Can compute probabilities of a given codeword for $\omega$ ...

(39)
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

- Can compute probabilities of a given codeword for $\omega$ ...

\[
p(x^n(\omega)) = \prod_{i=1}^{n} p(x_i(\omega)), \quad \omega \in \{1, \ldots, M\}
\] (39)
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

- Can compute probabilities of a given codeword for $\omega$ ...

$$p(x^n(\omega)) = \prod_{i=1}^{n} p(x_i(\omega)), \; \omega \in \{1, \ldots, M\}$$  \hspace{1cm} (39)

- ...or even the entire codebook:

$$p(C) = 2^{nR} \prod_{\omega=1}^{M} \prod_{i=1}^{n} p(x_i(\omega))$$  \hspace{1cm} (40)
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

- Can compute probabilities of a given codeword for $\omega$ ...

$$p(x^n(\omega)) = \prod_{i=1}^{n} p(x_i(\omega)), \ \omega \in \{1, \ldots, M\}$$  \hspace{1cm} (39)

- ...or even the entire codebook:

$$p(C) = \prod_{\omega=1}^{2^{nR}} \prod_{i=1}^{n} p(x_i(\omega))$$  \hspace{1cm} (40)
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Consider the following encoding/decoding scheme:

1. Generate a random codebook as above according to $p(x)$
2. Codebook known to both sender/receiver (who also knows $p(y|x)$).
3. Generate messages $W$ according to the uniform distribution (we’ll see why shortly), $p(W = \omega) = 2^{-nR}$, for $\omega = 1, \ldots, 2^{nR}$.
4. Send $x^n(\omega)$ over the channel.
5. Receiver receives $Y^n$ according to distribution

\[
Y^n \sim p(y^n|x^n(\omega)) = \prod_{i=1}^{n} p(y_i|x_i(\omega))
\]  

(41)

6. The signal is decoded using typical set decoding (to be described).
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Typical set decoding: Decode message as $\hat{\omega}$ if

\[ \text{Typical set decoding: } \text{Decode message as } \hat{\omega} \text{ if} \]

...
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Typical set decoding: Decode message as $\hat{\omega}$ if

1. $(x^n(\hat{\omega}), y^n)$ is jointly typical
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Typical set decoding: Decode message as $\hat{\omega}$ if

1. $(x^n(\hat{\omega}), y^n)$ is jointly typical
2. $\exists$ no other $k$ s.t. $(x^n(k), y^n) \in A^{(n)}_c$ (i.e., $\hat{\omega}$ is unique)
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Typical set decoding: Decode message as $\hat{\omega}$ if

1. $(x^n(\hat{\omega}), y^n)$ is jointly typical
2. $\exists$ no other $k$ s.t. $(x^n(k), y^n) \in A_{\epsilon}^{(n)}$ (i.e., $\hat{\omega}$ is unique)

Otherwise output special invalid integer “0” (error).
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Typical set decoding: Decode message as $\hat{\omega}$ if

1. $(x^n(\hat{\omega}), y^n)$ is jointly typical
2. $\exists$ no other $k$ s.t. $(x^n(k), y^n) \in A^n_\epsilon$ (i.e., $\hat{\omega}$ is unique)

Otherwise output special invalid integer “0” (error). Three types of errors might occur.
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Typical set decoding: Decode message as $\hat{\omega}$ if

1. $(x^n(\hat{\omega}), y^n)$ is jointly typical
2. $\exists$ no other $k$ s.t. $(x^n(k), y^n) \in A^{(n)}_{\epsilon}$ (i.e., $\hat{\omega}$ is unique)

Otherwise output special invalid integer “0” (error). Three types of errors might occur.

A: $\exists k \neq \hat{\omega}$ s.t. $(x^n(k), y^n) \in A^{(n)}_{\epsilon}$ (i.e., $> 1$ possible typical pair).
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Typical set decoding: Decode message as $\hat{\omega}$ if

1. $(x^n(\hat{\omega}), y^n)$ is jointly typical
2. $\exists$ no other $k$ s.t. $(x^n(k), y^n) \in A^{(n)}_\epsilon$ (i.e., $\hat{\omega}$ is unique)

Otherwise output special invalid integer “0” (error). Three types of errors might occur.

A: $\exists k \neq \hat{\omega}$ s.t. $(x^n(k), y^n) \in A^{(n)}_\epsilon$ (i.e., $> 1$ possible typical pair).
B: no $\hat{\omega}$ s.t. $(x^n(\hat{\omega}), y^n)$ is jointly typical.
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Typical set decoding: Decode message as $\hat{\omega}$ if

1. $(x^n(\hat{\omega}), y^n)$ is jointly typical
2. $\exists$ no other $k$ s.t. $(x^n(k), y^n) \in A_{c}^{(n)}$ (i.e., $\hat{\omega}$ is unique)

Otherwise output special invalid integer “0” (error). Three types of errors might occur.

A: $\exists k \neq \hat{\omega}$ s.t. $(x^n(k), y^n) \in A_{c}^{(n)}$ (i.e., $> 1$ possible typical pair).
B: no $\hat{\omega}$ s.t. $(x^n(\hat{\omega}), y^n)$ is jointly typical.
C: if $\hat{\omega} \neq \omega$, i.e., wrong codeword is jointly typical.

...
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Typical set decoding: Decode message as $\hat{\omega}$ if

1. $(x^n(\hat{\omega}), y^n)$ is jointly typical
2. $\exists$ no other $k$ s.t. $(x^n(k), y^n) \in A^{(n)}_\epsilon$ (i.e., $\hat{\omega}$ is unique)

Otherwise output special invalid integer “0” (error). Three types of errors might occur.

A: $\exists k \neq \hat{\omega}$ s.t. $(x^n(k), y^n) \in A^{(n)}_\epsilon$ (i.e., $> 1$ possible typical pair).

B: no $\hat{\omega}$ s.t. $(x^n(\hat{\omega}), y^n)$ is jointly typical.

C: if $\hat{\omega} \neq \omega$, i.e., wrong codeword is jointly typical.

Note: maximum likelihood decoding is optimal, but typical set decoding is not, but this will be good enough to show the result.
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

(also) three types of quality measures we might be interested in.
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

(also) three types of quality measures we might be interested in.

1. Code specific error

$$P_e^{(n)}(C) = \Pr(\hat{\omega} \neq \omega | C) = \frac{1}{2^{nR}} \sum_{i=1}^{2^{nR}} \lambda_i$$ (42)

where (as a reminder)

$$\lambda_i = \Pr(g(y^n) \neq i | X^n = x^n(i)) = \sum_{y^n} p(y^n | x^n(i)) 1\{g(y^n) \neq i\}$$
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

(also) three types of quality measures we might be interested in.

1. Code specific error

\[ P_{e(n)}(C) = \Pr(\hat{\omega} \neq \omega | C) = \frac{1}{2nR} \sum_{i=1}^{2nR} \lambda_i \] (42)

where (as a reminder)

\[ \lambda_i = \Pr(g(y^n) \neq i | X^n = x^n(i)) = \sum_{y^n} p(y^n | x^n(i)) \mathbf{1}\{g(y^n) \neq i\} \]

but we would like something easier to analyze.

\[ \text{Average error over all randomly generated codes (avg. of avg.)} \]

\[ \Pr(E) = \sum_C \Pr(C) \Pr(\hat{W} \neq W | C) = \sum_C \Pr(C) \, P_e(C) \] (43)
All rates \( R < C \) are achievable.

Proof that all rates \( R < C \) are achievable.

(also) three types of quality measures we might be interested in.

1. Code specific error

\[
P_{e}^{(n)}(C) = \Pr(\hat{\omega} \neq \omega | C) = \frac{1}{2^{nR}} \sum_{i=1}^{2^{nR}} \lambda_{i}
\]  

where (as a reminder)

\[
\lambda_{i} = \Pr(g(y^{n}) \neq i | X^{n} = x^{n}(i)) = \sum_{y^{n}} p(y^{n} | x^{n}(i)) \mathbf{1}\{g(y^{n}) \neq i\}
\]

but we would like something easier to analyze.

2. Average error over all randomly generated codes (avg. of avg.)

\[
\Pr(\mathcal{E}) = \sum_{C} \Pr(C) \Pr(\hat{W} \neq W | C) = \sum_{C} \Pr(C) P_{e}(C)
\]  

\[
\ldots
\]
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

(also) three types of quality measures we might be interested in.

1. **Code specific error**

   $$P_e^{(n)}(C) = \Pr(\hat{\omega} \neq \omega|C) = \frac{1}{2^{nR}} \sum_{i=1}^{2^{nR}} \lambda_i$$  \hspace{1cm} (42)

   where (as a reminder)
   $$\lambda_i = \Pr(g(y^n) \neq i|X^n = x^n(i)) = \sum_{y^n} p(y^n|x^n(i)) \mathbf{1}\{g(y^n) \neq i\}$$

   but we would like something easier to analyze.

2. **Average error over all randomly generated codes (avg. of avg.)**

   $$\Pr(\mathcal{E}) = \sum_C \Pr(C)\Pr(\hat{W} \neq W|C) = \sum_C \Pr(C)P_e(C)$$  \hspace{1cm} (43)

   Surprisingly, this is much easier to analyze than $P_e$...
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

(also) three types of quality measures we might be interested in.
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

(Also) three types of quality measures we might be interested in.

1. Max error of the code, ultimately what we want to use

$$P_{C,\text{max}}(C) = \max_{i \in \{1, 2, \ldots, M\}} \lambda_i$$

(44)

We want to show that if $R < C$, then exists a codebook $C$ s.t. this error $\to 0$ (and that if $R > C$ error must $\to 1$).
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

(also) three types of quality measures we might be interested in.

3. Max error of the code, ultimately what we want to use

$$P_{C,\text{max}}(C) = \max_{i \in \{1,2,\ldots,M\}} \lambda_i$$ (44)

We want to show that if $R < C$, then exists a codebook $C$ s.t. this error $\to 0$ (and that if $R > C$ error must $\to 1$).
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

(also) three types of quality measures we might be interested in.

1. Max error of the code, ultimately what we want to use

$$P_{C, \text{max}}(C) = \max_{i \in \{1, 2, \ldots, M\}} \lambda_i$$  \hspace{1em} (44)

We want to show that if $R < C$, then exists a codebook $C$ s.t. this error $\to 0$ (and that if $R > C$ error must $\to 1$).

Our method is to:

1. Expand average error (bullet 2 above) and show that it is small.

...
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

(also) three types of quality measures we might be interested in.

Max error of the code, ultimately what we want to use

$$P_{C, \text{max}}(C) = \max_{i \in \{1, 2, \ldots, M\}} \lambda_i$$ \hspace{1cm} (44)

We want to show that if $R < C$, then exists a codebook $C$ s.t. this error $\to 0$ (and that if $R > C$ error must $\to 1$).

Our method is to:

1. Expand average error (bullet 2 above) and show that it is small.

2. deduce that $\exists$ at least 1 code with small error
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

(also) three types of quality measures we might be interested in.

3. Max error of the code, ultimately what we want to use

$$PC_{\text{max}}(C) = \max_{i \in \{1,2,...,M\}} \lambda_i$$ (44)

We want to show that if $R < C$, then exists a codebook $C$ s.t. this error $\to 0$ (and that if $R > C$ error must $\to 1$).

Our method is to:

1. Expand average error (bullet 2 above) and show that it is small.
2. deduce that $\exists$ at least 1 code with small error
3. show that this can be modified to have small maximum probability of error.
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

$$\Pr(\mathcal{E}) = \sum_{\mathcal{C}} \Pr(\mathcal{C}) P_e^n(\mathcal{C})$$

(46)

$$\text{...}$$
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

$$\Pr(\mathcal{E}) = \sum_C \Pr(C) P_e^n(C) = \sum_C \Pr(C) \frac{1}{2nR} \sum_{\omega=1}^{2nR} \lambda_\omega(C)$$  \hspace{1cm} (45)

$$\text{but } \sum_C \Pr(C) \lambda_\omega(C) = \sum_C \Pr(C) \lambda(C)$$  \hspace{1cm} (46)

$$\text{stuff = } \sum \text{stuff}$$  \hspace{1cm} (47)

...
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

\[
\Pr(\mathcal{E}) = \sum_{C} \Pr(C) P_{e}^{(n)}(C) = \sum_{C} \Pr(C) \frac{1}{2^{nR}} \sum_{\omega=1}^{2^{nR}} \lambda_{\omega}(C) \tag{45}
\]

\[
= \frac{1}{2^{nR}} \sum_{\omega=1}^{2^{nR}} \sum_{C} \Pr(C) \lambda_{\omega}(C) \tag{46}
\]

\[
\tag{47}
\]

...
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

$$ \Pr(\mathcal{E}) = \sum_C \Pr(C) P_e^{(n)}(C) = \sum_C \Pr(C) \frac{1}{2^{nR}} \sum_{\omega=1}^{2^{nR}} \lambda_\omega(C) $$ (45)

$$ = \frac{1}{2^{nR}} \sum_{\omega=1}^{2^{nR}} \sum_C \Pr(C) \lambda_\omega(C) $$ (46)

but

...
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

\[
\Pr(\mathcal{E}) = \sum_C \Pr(C)P_{e(n)}(C) = \sum_C \Pr(C) \frac{1}{2^R} \sum_{\omega=1}^{2^R} \lambda_\omega(C)
\]

(45)

\[
= \frac{1}{2^R} \sum_{\omega=1}^{2^R} \sum_C \Pr(C) \lambda_\omega(C)
\]

(46)

but

\[
\sum_C \Pr(C) \lambda_\omega(C)
\]

(47)
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

$$\Pr(\mathcal{E}) = \sum_{C} \Pr(C) P_{e}^{(n)}(C) = \sum_{C} \Pr(C) \frac{1}{2nR} \sum_{\omega=1}^{2nR} \lambda_{\omega}(C)$$

(45)

$$= \frac{1}{2nR} \sum_{\omega=1}^{2nR} \sum_{C} \Pr(C) \lambda_{\omega}(C)$$

(46)

but

$$\sum_{C} \Pr(C) \lambda_{\omega}(C) = \sum_{C} \Pr(g(Y^{n}) \neq \omega | X^{n} = x^{n}(\omega)) \Pr(x^{n}(1), \ldots, x^{n}(2^{nR}))$$

(47)

...
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

\[
\Pr(\mathcal{E}) = \sum_{C} \Pr(C) P_{e}^{(n)}(C) = \sum_{C} \Pr(C) \frac{1}{2^{nR}} \sum_{\omega=1}^{2^{nR}} \lambda_{\omega}(C) \quad (45)
\]

\[
= \frac{1}{2^{nR}} \sum_{\omega=1}^{2^{nR}} \sum_{C} \Pr(C) \lambda_{\omega}(C) \quad (46)
\]

but

\[
\sum_{C} \Pr(C) \lambda_{\omega}(C) = \sum_{C} \Pr(g(Y^{n}) \neq \omega | X^{n} = x^{n}(\omega)) \Pr(x^{n}(1), \ldots, x^{n}(2^{nR}))
\]

\[
= \sum_{x^{n}(1), x^{n}(2), \ldots, x^{n}(2^{nR})} \text{stuff} \quad (47)
\]

\[\ldots\]
Proof that all rates $R < C$ are achievable.

\[
\sum_C \Pr(C) \lambda_\omega(C) \quad (48)
\]

\[
= \sum_{x^n(1), \ldots, x^n(\omega-1), x^n(\omega+1), \ldots, x^n(2nR)} \prod_{i \neq \omega} \Pr(x^n(i)) \quad (49)
\]

\[
= \sum_{x^n(\omega)} \Pr(g(Y^n) \neq \omega | X^n = x^n(\omega)) \Pr(x^n(\omega))
\]

\[
= \sum_{x^n \in X^n} \Pr(g(Y^n) \neq 1 | X^n = x^n(1)) \Pr(x^n(1)) = \sum_C \Pr(C) \lambda_1(C) = \beta \quad (50)
\]
All rates $R < C$ are achievable.

**Proof that all rates $R < C$ are achievable.**

\[
\sum_C \Pr(C) \lambda_\omega(C) 
\]

\[
= \sum \left\{ \prod_{i \neq \omega} \Pr(x^n(i)) \right\} 
\]

\[
= \sum p\left(\frac{x^n(1),...,x^n(\omega-1),x^n(\omega+1),...,x^n(2nR)}{x^n(\omega)}\right) \sum \Pr(g(Y^n) \neq \omega | X^n = x^n(\omega)) \Pr(x^n(\omega)) 
\]

\[
= \sum \Pr(g(Y^n) \neq \omega | X^n = x^n(\omega)) \Pr(x^n(\omega)) 
\]

\[
= \sum \Pr(g(Y^n) \neq 1 | X^n = x^n(1)) \Pr(x^n(1)) = \sum \Pr(C) \lambda_1(C) = \beta 
\]

Last sum is same regardless of $\omega$, call it $\beta$. Thus, we can can arbitrarily assume that $\omega = 1$. 


All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

So we get

$$\Pr(\mathcal{E}) = \sum_C \Pr(C) P_{e}^{(n)}(C) = \frac{1}{2^{nR}} \sum_{\omega=1}^{2^{nR}} \beta == \sum_C \Pr(C) \lambda_1(C) = \Pr(\mathcal{E}|W=1)$$  

(51)
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

So we get

$$
\Pr(\mathcal{E}) = \sum_{C} \Pr(C) P_{\epsilon}^{(n)}(C) = \frac{1}{2nR} \sum_{\omega=1}^{2^{nR}} \beta = \sum_{C} \Pr(C) \lambda_1(C) = \Pr(\mathcal{E}|W = 1) \tag{51}
$$

Next, define the random events:

$$
E_i \triangleq \left\{ (x^n(i), y^n) \in A^{(n)}_{\epsilon} \right\} \text{ for } i = 1, \ldots, 2^{nR} \tag{52}
$$
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

So we get

$$\Pr(\mathcal{E}) = \sum_C \Pr(C) P_{\epsilon}^{(n)}(C) = \frac{1}{2^{nR}} \sum_{\omega=1}^{2^{nR}} \beta = \sum_C \Pr(C) \lambda_1(C) = \Pr(\mathcal{E}|W = 1)$$

(51)

- Next, define the random events:

$$E_i \triangleq \left\{ (x^n(i), y^n) \in A_\epsilon^{(n)} \right\} \text{ for } i = 1, \ldots, 2^{nR}$$

(52)

- Assume that input is $x^n(1)$ (i.e., first message sent).
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

So we get

$$
\Pr(\mathcal{E}) = \sum_C \Pr(C) P_{e}^{(n)}(C) = \frac{1}{2^{nR}} \sum_{\omega=1}^{2^{nR}} \beta = \sum_C \Pr(C) \lambda_1(C) = \Pr(\mathcal{E}|W = 1) 
$$

(51)

- Next, define the random events:

$$
E_i \triangleq \left\{ (x^n(i), y^n) \in A_c^{(n)} \right\} \text{ for } i = 1, \ldots, 2^{nR} 
$$

(52)

- Assume that input is $x^n(1)$ (i.e., first message sent).
- Then the no error event is the same as: $E_1 \cap \lnot (E_2 \cup E_3 \cup \cdots \cup E_M)$. 

...
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Various flavors of error

- $E_1^c$ means that the transmitted and received codeword are not jointly typical (this is error type B from before).
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Various flavors of error

- $E_1^c$ means that the transmitted and received codeword are not jointly typical (this is error type B from before).
- $E_2 \cup E_3 \cup \cdots \cup E_{2nR}$. This either:
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Various flavors of error

- $E_1^C$ means that the transmitted and received codeword are not jointly typical (this is error type B from before).
- $E_2 \cup E_3 \cup \cdots \cup E_{2nR}$. This either:
  - Type C: wrong codeword is jointly typical with received sequence
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Various flavors of error

- $E_1^C$ means that the transmitted and received codeword are not jointly typical (this is error type B from before).
- $E_2 \cup E_3 \cup \cdots \cup E_{2nR}$. This either:
  - Type C: wrong codeword is jointly typical with received sequence
  - Type A: greater than 1 codeword is jointly typical with received sequence
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Various flavors of error

- $E_1^C$ means that the transmitted and received codeword are not jointly typical (this is error type B from before).
- $E_2 \cup E_3 \cup \cdots \cup E_{2nR}$. This either:
  - Type C: wrong codeword is jointly typical with received sequence
  - Type A: greater than 1 codeword is jointly typical with received sequence

so this is type $C$ and $A$ both.
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Various flavors of error

- $E_1^C$ means that the transmitted and received codeword are not jointly typical (this is error type B from before).
- $E_2 \cup E_3 \cup \cdots \cup E_{2nR}$. This either:
  - Type C: wrong codeword is jointly typical with received sequence
  - Type A: greater than 1 codeword is jointly typical with received sequence

so this is type $C$ and $A$ both.
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Various flavors of error

- $E_1^C$ means that the transmitted and received codeword are not jointly typical (this is error type B from before).
- $E_2 \cup E_3 \cup \cdots \cup E_{2nR}$. This either:
  - Type C: wrong codeword is jointly typical with received sequence
  - Type A: greater than 1 codeword is jointly typical with received sequence

so this is type $C$ and $A$ both.

Our goal is to bound the probability of error, but lets look at some figures first.
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable. 

set of all jointly typical pairs of sequences $2^{nH(X,Y)}$

set of all pairs of marginally typical sequences

$2^{nH(Y|X)}$

$2^{nH(Y)}$

$2^{nH(X)}$

$2^{nH(X|Y)}$
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Vertical axis is lexicographic order of possible codewords.

Subset selection of the $2^{nR}$ random $\mathcal{X}^n$ codewords (chosen by the random selection procedure) for $i = 1, 2, \ldots, M$. Here, $2^{nR} = M = 4$.

Dots are the jointly typical sequences.
Proof that all rates $R < C$ are achievable.

- $y(a)$, on sending $x(*)$
- $y(d)$, on sending $x(1)$
- $y(b)$, on sending $x(4)$
- $y(c)$, on sending $x(1)$

$X^n$ with $x(1)$, $x(3)$, $x(4)$, $x(2)$

$y(a)$ not jointly typical with any of the sent codewords.
Error type B

$g(y(a)) = 0$

$y(d)$ should not be jointly typical with $x(4)$ but it is. Wrong jointly typical sequence.
Error type C.

$g(y(d)) = 4$

$y(b)$ is jointly typical only with $x(4)$, so no error

$g(y(b)) = 4$

$y(c)$ is jointly typical with both $x(1)$ and $x(3)$, so
Error type A

$g(y(c)) = 0$

$E_1^c$ $E_2 \cup E_3 \cup \ldots$
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Goal: bound the probability of error:

\[
\Pr(\mathcal{E}|W = 1) = \Pr(E_1^c \cup E_2 \cup E_3 \ldots) 
\]

\[
\leq \Pr(E_1^c) + \sum_{i=2}^{2^n R} \Pr(E_i) \text{ by the union bound} 
\]

We have that

\[
\Pr(E_1^c) = \Pr(A_{\epsilon}^{(n)c}) \to 0 \text{ as } n \to \infty 
\]

So, $\forall \epsilon, \exists n_0$ s.t.

\[
\Pr(E_1^c) \leq \epsilon, \ \forall n > n_0 
\]
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

Also, because of random code generation process (and recall, $W = 1$)

$$X^n(1) \perp \perp X^n(i) \Rightarrow Y^n \perp \perp X^n(i), \text{ for } i \neq 1$$  \hspace{1cm} (57)

this gives

$$\Pr((X^n(i), Y^n) \in A^{(n)}_{\epsilon}) \leq 2^{-n(I(X;Y)-3\epsilon)}$$  \hspace{1cm} (58)

by the joint AEP.
Proof that all rates $R < C$ are achievable.

So we get:

$$\Pr(\mathcal{E}) = \Pr(\mathcal{E}|W = 1) \leq \Pr(E_1^c) + \sum_{i=2}^{2^nR} \Pr(E_i)$$  \hspace{1cm} (59)

$$\leq \epsilon + \sum_{i=2}^{2^nR} 2^{-n(I(X;Y) - 3\epsilon)}$$  \hspace{1cm} (60)

$$= \epsilon + (2^{nR} - 1)2^{-n(I(X;Y) - 3\epsilon)}$$  \hspace{1cm} (61)

$$\leq \epsilon + 2^{3n\epsilon}2^{-n(I(X;Y) - R)}$$  \hspace{1cm} (62)

$$= \epsilon + 2^{-n((I(X;Y) - 3\epsilon) - R)}$$  \hspace{1cm} (63)

$$\leq 2\epsilon \quad \text{for large enough } n$$  \hspace{1cm} (64)

The last statement is true only if $I(X;Y) - 3\epsilon > R$. …
Proof that all rates $R < C$ are achievable.

- So if we chose $R < I(X; Y)$ (strictly), we can find an $\epsilon$ and $n$ so that the average probability of error $Pr(\mathcal{E}) \leq 2\epsilon$, can be made as small as we want.
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

- So if we chose $R < I(X; Y)$ (strictly), we can find an $\epsilon$ and $n$ so that the average probability of error $Pr(\mathcal{E}) \leq 2\epsilon$, can be made as small as we want.

- But, we need to get from an average to a max probability of error, and bound that.

...
All rates \( R < C \) are achievable.

Proof that all rates \( R < C \) are achievable.

- So if we chose \( R < I(X; Y) \) (strictly), we can find an \( \epsilon \) and \( n \) so that the average probability of error \( \Pr(\mathcal{E}) \leq 2\epsilon \), can be made as small as we want.

- But, we need to get from an average to a max probability of error, and bound that.

- First, choose \( p^*(x) = \arg\max_{p(x)} I(X; Y) \) rather than uniform \( p(x) \), to change the condition from \( R < I(X; Y) \) to \( R < C \). Thus, This gives us higher rate limit.
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

- So if we chose $R < I(X; Y)$ (strictly), we can find an $\epsilon$ and $n$ so that the average probability of error $Pr(\mathcal{E}) \leq 2\epsilon$, can be made as small as we want.

- But, we need to get from an average to a max probability of error, and bound that.

- First, choose $p^*(x) = \arg\max_{p(x)} I(X; Y)$ rather than uniform $p(x)$, to change the condition from $R < I(X; Y)$ to $R < C$. Thus, This gives us higher rate limit.

- If $Pr(\mathcal{E}) \leq 2\epsilon$, the bound on the average error is small, so there must exist some specific code, say $C^*$ s.t.

$$P_e^{(n)}(C^*) \leq 2\epsilon$$  \hfill (65)

...
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.

next time, we’ll see how to do this ...
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.
All rates $R < C$ are achievable.

Proof that all rates $R < C$ are achievable.