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.
Announcements, Assignments, and Reminders

- Reminder: Homework 3 due next Sunday (Feb 5th) at 5:00pm our dropbox (https://catalyst.uw.edu/collectit/dropbox/karna/19164).

- This homework is long! Don’t wait to start it!

- 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

- L1 (1/3): Overview, Entropy
- L2 (1/5): Props. Entropy, Mutual Information, KL-Divergence
- L3 (1/10): KL-Divergence, Jensen, properties, Data Proc. Inequality
- L5 (1/17): Fano, AEP
- L6 (1/19): snow
- L6 (1/24): AEP, source coding
- L7 (1/26): Method of Types
- L9 (2/2):
- L11 (2/7):
- L12 (2/9):
- L13 (2/14): Midterm
- L14 (2/16):
- L15 (2/21):
- L16 (2/23):
- L175 (2/28):
- L18 (3/1):
- L19 (3/6):
- L20 (3/8):

Finals Week: March 12th–16th.
Definition: the “type” of the sample

Let \( X_1, X_2, \ldots, X_n \equiv X_{1:n} \) be a length-\( n \) sample of a D-ary discrete random variable. So \( x_i \in \mathcal{X} \) and alphabet size \( D = |\mathcal{X}| \), and \( \mathcal{X} = (a_1, a_2, \ldots, a_D) \).

Define a statistic which is the empirical histogram of this sample.

\[
P_{x_{1:n}} \triangleq \left( \frac{n(a_1|x_{1:n})}{n}, \frac{n(a_2|x_{1:n})}{n}, \ldots, \frac{n(a_D|x_{1:n})}{n} \right) \tag{1}
\]

where \( n(a_i|x_{1:n}) \) counts occurrence of symbol \( a_i \) in sample \( x_{1:n} \).

Define \( \mathcal{P}_n \) be the set of all possible types with denominator \( n \).

A given type is \( P \in \mathcal{P}_n \).

Type class, for \( P \in \mathcal{P}_n \), is \( T(P) \). I.e.,

\[
T(P) \triangleq \{ x_{1:n} \in \mathcal{X}^n : P_{x_{1:n}} = P \} \tag{2}
\]

set of all sequences of length \( n \) having a certain histogram \( P \).
Notational Summary

- For sequences of length $n$:
- the type (or histogram) of a sample $x_{1:n}$

$$P_{x_{1:n}} \triangleq \left( \frac{n(a_1|x_{1:n})}{n}, \frac{n(a_2|x_{1:n})}{n}, \ldots, \frac{n(a_D|x_{1:n})}{n} \right); \quad (3)$$

- the set of all types (or histograms) $\mathcal{P}_n$;
- Some particular type $P \in \mathcal{P}_n$.
- For a given type $P$, the set of all sequences with that type $T(P)$. 

Division of set of all sequences into type classes

- \( \mathcal{P}_n = \{ P_1, P_2, \ldots, P_{|\mathcal{P}_n|} \} \) is the set of all types,
- Thus, \( \bigcup_{P \in \mathcal{P}_n} T(P) = \mathcal{X}^n \).
- The space of all sequences.

\( \mathcal{X}^n \) : the set of all sequences of length \( n \)
Division of set of all sequences into type classes

- $\mathcal{P}_n = \{P_1, P_2, \ldots, P_{|\mathcal{P}_n|}\}$ is the set of all types,
- Thus, $\bigcup_{P \in \mathcal{P}_n} T(P) = \mathcal{X}^n$.
- The space of all sequences.

$\mathcal{X}^n$: partitioned into blocks within which all sequences have the same type.

![Diagram](image-url)
Bound on number of type classes

Proposition 2.1

\[ |\mathcal{P}_n| \leq (n + 1)|\mathcal{X}| \]
Bound on number of type classes

Proposition 2.1

\[ |\mathcal{P}_n| \leq (n + 1)|\mathcal{X}| \]
Probability depends only on the type

Theorem 2.2

- Let $X_1, X_2, \ldots, X_n$ be i.i.d. $\sim Q(x)$,
- with extension $Q^n(x_1:n) = \prod_i Q(x_i)$, with $Q$ otherwise arbitrary.
- The probability of the sequence depends only on the type
- restated, the probability is “independent” of the sequence given the type and $Q$
- That is

$$Q^n(x_1:n) = 2^{-n\left[H(P_{x_1:n}) + D(P_{x_1:n} \| Q)\right]}$$ (5)

- So, probability doesn’t depend on the sequence, once we are given the type
- Compare with sufficient statistics
- all sequences with the same type have the same probability.
Probability depends only on the type

- Corollary: If $Q$ is a rational distribution (i.e., a possible type) and if $x_{1:n} \in T(Q)$, then
  \[ Q^n(x_{1:n}) = 2^{-nH(Q)} \]  
  a result already familiar to us.

- What if $Q$ was irrational? Intuition: we could make $D(P_{x_{1:n}} || Q)$ as small as we want, if we make $n$ large.
Proposition 2.3

For any type $P \in \mathcal{P}_n$, we have

$$\frac{1}{(n + 1)|\mathcal{X}|} 2^{nH(P)} \leq |T(P)| \leq 2^{nH(P)}$$

(7)
How probable is each type class?

- Notation: $a_n \asymp b_n$ if $\lim_{n \to \infty} \frac{1}{n} \log \frac{a_n}{b_n} = 0$. 
How probable is each type class?

- Notation: \( a_n \overset{!}{=} b_n \) if \( \lim_{n \to \infty} \frac{1}{n} \log \frac{a_n}{b_n} = 0 \).

**Theorem 2.4**

For any \( P \in \mathcal{P}_n \), and any distribution \( Q \), the probability of type class \( T(P) \) under \( Q^n \) is such that \( Q^n(T(P)) \doteq 2^{-nD(P\|Q)} \). Specifically,

\[
\frac{1}{(n+1)|X|} 2^{-nD(P\|Q)} \leq Q^n(T(P)) \leq 2^{-nD(P\|Q)} \quad (8)
\]

Note: so any type less close than the “closest” type to \( Q \) will decrease in probability exponentially (in \( n \)) faster than the most probable type.
How probable is each type class?

- Notation: \( a_n \doteq b_n \) if \( \lim_{n \to \infty} \frac{1}{n} \log \frac{a_n}{b_n} = 0 \).

**Theorem 2.4**

For any \( P \in \mathcal{P}_n \), and any distribution \( Q \), the probability of type class \( T(P) \) under \( Q^n \) is such that \( Q^n(T(P)) \doteq 2^{-nD(P||Q)} \). Specifically,

\[
\frac{1}{(n + 1)|\mathcal{X}|} 2^{-nD(P||Q)} \leq Q^n(T(P)) \leq 2^{-nD(P||Q)} \quad (8)
\]

Note: so any type less close than the “closest” type to \( Q \) will decrease in probability exponentially (in \( n \)) faster than the most probable type.
Summary of basic theorems

- Number of types with denominator \(n\)
  \[
  |\mathcal{P}_n| \leq (n + 1)|\mathcal{X}|
  \]  
  \(9\)

- \(p(x_{1:n})\) depends only on the type (prob. indep. of sample given type)
  \[
  Q^n(x_{1:n}) = 2^{-n}[H(P_{x_{1:n}}) + D(P_{x_{1:n}} \| Q)]
  \]  
  \(10\)

- Size of the type class
  \[
  |T(P)| = 2^{nH(P)}
  \]  
  \(11\)

- Probability of a type class
  \[
  Q^n(T(P)) = 2^{-nD(P \| Q)}
  \]  
  \(12\)
Types with the most probability

- Q: Which types will have the most probability?
- A: Clearly, the ones that are closest to the true distribution.
- The property \( Q^n(T(P)) \doteq 2^{-nD(P\|Q)} \) says that the ones that are farther away will have exponentially smaller probability than the others, as \( n \to \infty \).
- This suggests that “typical set of sequences” applies here as well,
- in fact

**Definition 2.5 (typical set of sequences)**

Let \( X_1, X_2, \ldots, X_n \) be i.i.d. \( \forall i, x_i \sim Q(x) \). Then the typical set is defined as

\[
T_Q^\epsilon = \{ x_{1:n} : D(P_{x_{1:n}}\|Q) \leq \epsilon \}
\]  

(13)

- Intuitively, these are sequences that come from types that are \( \epsilon \)-close to \( Q \) in the KL-sense.
Theorem 2.6

Let $X_1, X_2, \ldots, X_n$ be i.i.d. $\forall i, x_i \sim Q(x)$. Then the probability of the complement of the typical set $\bar{T}_Q^\epsilon$ has expression:

$$Q(\bar{T}_Q^\epsilon) = Q(\{x_{1:n} : D(P_{x_{1:n}} \parallel Q) > \epsilon \}) \leq 2^{-n(\epsilon - |X| \frac{\log(n+1)}{n})}$$ (14)

and therefore,

$$D(P_{X_{1:n}} \parallel Q) \xrightarrow{p} 0 \text{ as } n \to \infty$$ (15)

- Intuitively, this means that types that are more than $\epsilon$ away from $Q$ have decreasing probability.
- Moreover, the typical set, which ends up for large $n$ being the only thing that occurs without vanishingly small probability, is such that the KL divergence gets between the type and $Q$ quickly gets arbitrarily small.
Universal Source Coding

- If we know $p(x)$, then we will be able to develop a code to compress sources generated by $p(x)$. Huffman, Lempel-Ziv, etc. are codes that, as we will soon see, do that.
Universal Source Coding

- If we know \( p(x) \), then we will be able to develop a code to compress sources generated by \( p(x) \). Huffman, Lempel-Ziv, etc. are codes that, as we will soon see, do that.

- What if we don't know \( p(x) \)?
Universal Source Coding

- If we know $p(x)$, then we will be able to develop a code to compress sources generated by $p(x)$. Huffman, Lempel-Ziv, etc. are codes that, as we will soon see, do that.

- What if we don't know $p(x)$?

- Q: do there exist codes that can compress without knowing $p(x)$ and that do so down to the entropy limit?
Universal Source Coding

- If we know \( p(x) \), then we will be able to develop a code to compress sources generated by \( p(x) \). Huffman, Lempel-Ziv, etc. are codes that, as we will soon see, do that.

- What if we don't know \( p(x) \)?

- Q: do there exist codes that can compress without knowing \( p(x) \) and that do so down to the entropy limit?

- Q: can we compress down to the rate \( R \) (in units of bits per source symbol) if \( R > H(Q) \)? (this is Shannon's source coding theorem)
Universal Source Coding

- If we know $p(x)$, then we will be able to develop a code to compress sources generated by $p(x)$. Huffman, Lempel-Ziv, etc. are codes that, as we will soon see, do that.

- What if we don't know $p(x)$?

- Q: do there exist codes that can compress without knowing $p(x)$ and that do so down to the entropy limit?

- Q: can we compress down to the rate $R$ (in units of bits per source symbol) if $R > H(Q)$? (this is Shannon’s source coding theorem)

- What happens if $R < H(Q)$? (this is the converse of Shannon’s source coding theorem)
Universal Source Coding

If we know $p(x)$, then we will be able to develop a code to compress sources generated by $p(x)$. Huffman, Lempel-Ziv, etc. are codes that, as we will soon see, do that.

What if we don't know $p(x)$?

Q: do there exist codes that can compress without knowing $p(x)$ and that do so down to the entropy limit?

Q: can we compress down to the rate $R$ (in units of bits per source symbol) if $R > H(Q)$? (this is Shannon's source coding theorem)

What happens if $R < H(Q)$? (this is the converse of Shannon's source coding theorem)

We'll formally prove this theorem using the method of types.
Universal Source Coding: intuitive idea from AEP

- Basic idea is similar to the typical set $A_{\varepsilon}^{(n)}$ we’ve already seen: when $n$ is long enough, the only sequences that occur (with non-vanishingly small probability) will be typical.
Universal Source Coding: intuitive idea from AEP

- Basic idea is similar to the typical set $A_c^{(n)}$ we’ve already seen: when $n$ is long enough, the only sequences that occur (with non-vanishingly small probability) will be typical.

- If we encounter such a sequence, it “must” be typical since the only things that occur are typical.
Universal Source Coding: intuitive idea from AEP

- Basic idea is similar to the typical set $A_{c}^{(n)}$ we’ve already seen: when $n$ is long enough, the only sequences that occur (with non-vanishingly small probability) will be typical.
- If we encounter such a sequence, it “must” be typical since the only things that occur are typical.
- Thus, we only encode the things we see, and we count them along the way.
Universal Source Coding: intuitive idea from AEP

- Basic idea is similar to the typical set $A_\epsilon^{(n)}$ we’ve already seen: when $n$ is long enough, the only sequences that occur (with non-vanishingly small probability) will be typical.
- If we encounter such a sequence, it “must” be typical since the only things that occur are typical.
- Thus, we only encode the things we see, and we count them along the way.
- In the end, we’ll need at most $|A_\epsilon^{(n)}| \leq 2^{nH}$ code words for which we can index with $nH$ bits.
Universal Source Coding: intuitive idea from AEP

- Basic idea is similar to the typical set $A_{\epsilon}^{(n)}$ we’ve already seen: when $n$ is long enough, the only sequences that occur (with non-vanishingly small probability) will be typical.
- If we encounter such a sequence, it “must” be typical since the only things that occur are typical.
- Thus, we only encode the things we see, and we count them along the way.
- In the end, we’ll need at most $|A_{\epsilon}^{(n)}|$ code words for which we can index with $nH$ bits.
- We want to formalize Shannon’s theorem and its converse using the method of types.
Universal Source Coding: intuitive idea from types

- There are $2^{nH(P)}$ sequences of type $P$. 

---

Prof. Jeff Bilmes  
Universal Source Coding: intuitive idea from types

- There are $2^{nH(P)}$ sequences of type $P$.
- Thus, we use $nH(P)$ bits to represent such sequences.
Universal Source Coding: intuitive idea from types

- There are $2^{nH(P)}$ sequences of type $P$.
- Thus, we use $nH(P)$ bits to represent such sequences.
- If $R > H(P)$, we are easily able to use $nR$ bits to represent such sequences.
Universal Source Coding: intuitive idea from types

- There are $2^{nH(P)}$ sequences of type $P$.
- Thus, we use $nH(P)$ bits to represent such sequences.
- If $R > H(P)$, we are easily able to use $nR$ bits to represent such sequences.
- As $n$ gets big, only the types $P$ that are “close” to $Q$ will occur.
Universal Source Coding: intuitive idea from types

- There are $2^{nH(P)}$ sequences of type $P$.
- Thus, we use $nH(P)$ bits to represent such sequences.
- If $R > H(P)$, we are easily able to use $nR$ bits to represent such sequences.
- As $n$ gets big, only the types $P$ that are “close” to $Q$ will occur.
- There are an exponential (in $n$) number of sequences.
Universal Source Coding: intuitive idea from types

- There are $2^{nH(P)}$ sequences of type $P$.
- Thus, we use $nH(P)$ bits to represent such sequences.
- If $R > H(P)$, we are easily able to use $nR$ bits to represent such sequences.
- As $n$ gets big, only the types $P$ that are “close” to $Q$ will occur.
- There are an exponential (in $n$) number of sequences.
- There are only a polynomial (in $n$) number of types.
Universal Source Coding: intuitive idea from types

- There are $2^{nH(P)}$ sequences of type $P$.
- Thus, we use $nH(P)$ bits to represent such sequences.
- If $R > H(P)$, we are easily able to use $nR$ bits to represent such sequences.
- As $n$ gets big, only the types $P$ that are “close” to $Q$ will occur.
- There are an exponential (in $n$) number of sequences.
- There are only a polynomial (in $n$) number of types.
- Thus, one type must eventually get “all” of the probability.
Universal Source Coding: intuitive idea from types

- There are $2^{nH(P)}$ sequences of type $P$.
- Thus, we use $nH(P)$ bits to represent such sequences.
- If $R > H(P)$, we are easily able to use $nR$ bits to represent such sequences.
- As $n$ gets big, only the types $P$ that are “close” to $Q$ will occur.
- There are an exponential (in $n$) number of sequences.
- There are only a polynomial (in $n$) number of types.
- Thus, one type must eventually get “all” of the probability.
- If $P \approx Q$, and $H(Q) < R$, then all types that actually “occur” can be represented in $R$ bits per source symbol.
Universal Source Coding: intuitive idea from types

- There are $2^{nH(P)}$ sequences of type $P$.
- Thus, we use $nH(P)$ bits to represent such sequences.
- If $R > H(P)$, we are easily able to use $nR$ bits to represent such sequences.
- As $n$ gets big, only the types $P$ that are “close” to $Q$ will occur.
- There are an exponential (in $n$) number of sequences.
- There are only a polynomial (in $n$) number of types.
- Thus, one type must eventually get “all” of the probability.
- If $P \approx Q$, and $H(Q) < R$, then all types that actually “occur” can be represented in $R$ bits per source symbol.
- If $P \approx Q$, and $H(Q) > R$, then types that occur can not be represented in $R$ bits per source symbol.
Our encoder setup

- Recall from earlier our $x$ to $y$ encoder setup.

\[ \{X_1, X_2, \ldots, X_n\} \]
\[ X_i \in \{a_1, a_2, \ldots, a_K\} \]
\[ K^n \text{ possible messages} \]
\[ n \text{ letters or alphabet symbols} \]

\[ \{Y_1, Y_2, \ldots, Y_m\} \]
\[ Y_i \in \{0, 1\} \]
\[ 2^m \text{ possible messages} \]
\[ m \text{ total bits} \]
(M, n) codes

- Fixed rate block code of rate $R$. 
\((M, n)\) codes

- Fixed rate block code of rate \(R\).
- There are \(M\) code words, \(M = \text{number of possible messages}\).
(M, n) codes

- Fixed rate block code of rate $R$.
- There are $M$ code words, $M =$ number of possible messages.
- There are $n$ source symbols encoded at a time in each code word.
Fixed rate block code of rate $R$.

There are $M$ code words, $M =$ number of possible messages.

There are $n$ source symbols encoded at a time in each code word.

An encoder maps from length-$n$ strings of source symbols to length-$m$ bit strings.
### $(M, n)$ codes

- **Fixed rate block code of rate** $R$.
- There are $M$ code words, $M =$ number of possible messages.
- There are $n$ source symbols encoded at a time in each code word.
- An encoder maps from length-$n$ strings of source symbols to length-$m$ bit strings.

The rate $R$ of the code depends on $M$ and $n$

\[
R = \frac{\log M}{n} = \frac{\log(\# \text{ of code words})}{\# \text{ of source symbols}} \tag{16}
\]
Fixed rate block code of rate $R$

- An $(M, n)$ code is one that uses $M$ code words for $n$ source symbols.
Fixed rate block code of rate $R$

- An $(M, n)$ code is one that uses $M$ code words for $n$ source symbols.
- Such a code thus has rate of $R = \frac{\log M}{n}$ bits per source symbol. So we need $\log M = nR$ bits to index this code.
Fixed rate block code of rate $R$

- An $(M, n)$ code is one that uses $M$ code words for $n$ source symbols.
- Such a code thus has rate of $R = \frac{\log M}{n}$ bits per source symbol. So we need $\log M = nR$ bits to index this code.
- Then a code is defined as follows:
Fixed rate block code of rate $R$

- An $(M, n)$ code is one that uses $M$ code words for $n$ source symbols.
- Such a code thus has rate of $R = \frac{\log M}{n}$ bits per source symbol. So we need $\log M = nR$ bits to index this code.
- Then a code is defined as follows:

**Definition 3.1 (fixed rate block code of rate $R$)**

Let $X_1, X_2, \ldots, X_n \sim Q$, i.i.d. but $Q$ unknown. We have encoder and decoder functions as follows:
Fixed rate block code of rate $R$

- An $(M, n)$ code is one that uses $M$ code words for $n$ source symbols.
- Such a code thus has rate of $R = \frac{\log M}{n}$ bits per source symbol. So we need $\log M = nR$ bits to index this code.
- Then a code is defined as follows:

**Definition 3.1 (fixed rate block code of rate $R$)**

Let $X_1, X_2, \ldots, X_n \sim Q$, i.i.d. but $Q$ unknown. We have encoder and decoder functions as follows:

Encoder: $f_n : \mathcal{X}^n \to \{1, 2, \ldots, 2^{nR}\}$  \hspace{1cm} (17)

(18)
Fixed rate block code of rate $R$

- An $(M, n)$ code is one that uses $M$ code words for $n$ source symbols.
- Such a code thus has rate of $R = \frac{\log M}{n}$ bits per source symbol. So we need $\log M = nR$ bits to index this code.
- Then a code is defined as follows:

**Definition 3.1 (fixed rate block code of rate $R$)**

Let $X_1, X_2, \ldots, X_n \sim Q$, i.i.d. but $Q$ unknown. We have encoder and decoder functions as follows:

Encoder: $f_n : \mathcal{X}^n \rightarrow \{1, 2, \ldots, 2^{nR}\}$

Decoder: $\phi_n : \{1, 2, \ldots, 2^{nR}\} \rightarrow \mathcal{X}^n$
Fixed rate block code of rate $R$

- An $(M, n)$ code is one that uses $M$ code words for $n$ source symbols.
- Such a code thus has rate of $R = \frac{\log M}{n}$ bits per source symbol. So we need $\log M = nR$ bits to index this code.
- Then a code is defined as follows:

Definition 3.1 (fixed rate block code of rate $R$)

Let $X_1, X_2, \ldots, X_n \sim Q$, i.i.d. but $Q$ unknown. We have encoder and decoder functions as follows:

Encoder: $f_n : \mathcal{X}^n \rightarrow \{1, 2, \ldots, 2^{nR}\}$

Decoder: $\phi_n : \{1, 2, \ldots, 2^{nR}\} \rightarrow \mathcal{X}^n$

and probability of error

$$P_e^{(n)} = Q^n(\{x_{1:n} : \phi_n(f_n(x_{1:n})) \neq x_{1:n}\})$$
Fixed rate block code of rate $R$

- An $(M, n)$ code is one that uses $M$ code words for $n$ source symbols.
- Such a code thus has rate of $R = \frac{\log M}{n}$ bits per source symbol. So we need $\log M = nR$ bits to index this code.
- Then a code is defined as follows:

**Definition 3.1 (fixed rate block code of rate $R$)**

Let $X_1, X_2, \ldots, X_n \sim Q$, i.i.d. but $Q$ unknown. We have encoder and decoder functions as follows:

Encoder: $f_n : X^n \rightarrow \{1, 2, \ldots, 2^{nR}\}$  \hspace{1cm} (17)

Decoder: $\phi_n : \{1, 2, \ldots, 2^{nR}\} \rightarrow X^n$  \hspace{1cm} (18)

and probability of error

$$P_{e}^{(n)} = Q^n(\{x_{1:n} : \phi_n(f_n(x_{1:n})) \neq x_{1:n}\})$$  \hspace{1cm} (19)

- Notation: $(M, n) = (2^{nR}, n)$ designates a series (in $n$) of such codes.
Definition 3.2 (Universal rate $R$ block code)

A rate $R$ block code for a source is universal if the functions $f_n$ and $\phi_n$ do not depend on the source distribution $Q$ and if

$$P_e^{(n)} \rightarrow 0 \text{ as } n \rightarrow \infty \text{ whenever } H(Q) < R \quad (20)$$
Definition 3.2 (Universal rate $R$ block code)

A rate $R$ block code for a source is universal if the functions $f_n$ and $\phi_n$ do not depend on the source distribution $Q$ and if

$$P_e^{(n)} \rightarrow 0 \text{ as } n \rightarrow \infty \text{ whenever } H(Q) < R$$  \hspace{1cm} (20)

- So we require the “ability to code” at rate $R$, which really means code without error, or the error goes to zero for larger block length.
Definition 3.2 (Universal rate $R$ block code)

A rate $R$ block code for a source is universal if the functions $f_n$ and $\phi_n$ do not depend on the source distribution $Q$ and if

$$P_e^{(n)} \to 0 \text{ as } n \to \infty \text{ whenever } H(Q) < R$$

(20)

- So we require the “ability to code” at rate $R$, which really means code without error, or the error goes to zero for larger block length.
- We next state and prove one of Shannon’s main theorems.
Universal Code

Definition 3.2 (Universal rate $R$ block code)

A rate $R$ block code for a source is universal if the functions $f_n$ and $\phi_n$ do not depend on the source distribution $Q$ and if

$$P_e^{(n)} \to 0 \text{ as } n \to \infty \text{ whenever } H(Q) < R \tag{20}$$

- So we require the “ability to code” at rate $R$, which really means code without error, or the error goes to zero for larger block length.
- We next state and prove one of Shannon’s main theorems.
- If $R > H(Q)$, then there exists a sequence (in $n$) of codes with the error of becoming vanishingly small.
Universal Code

Definition 3.2 (Universal rate $R$ block code)

A rate $R$ block code for a source is universal if the functions $f_n$ and $\phi_n$ do not depend on the source distribution $Q$ and if

$$P_e^{(n)} \to 0 \text{ as } n \to \infty \text{ whenever } H(Q) < R$$

(20)

- So we require the “ability to code” at rate $R$, which really means code without error, or the error goes to zero for larger block length.
- We next state and prove one of Shannon’s main theorems.
- If $R > H(Q)$, then there exists a sequence (in $n$) of codes with the error of becoming vanishingly small.
- Conversely, if $R < H(Q)$, then the error goes to 1.
Source Coding Theorem

**Theorem 3.3 (Shannon’s Source Coding Theorem)**

∃ a sequence $(2^{nR}, n)$ of universal source codes such that $P_e^{(n)} \to 0$ for all source distributions $Q$ such that $H(Q) < R$.

**Proof.**

- Fix $R > H(Q)$ to be strictly greater than entropy.
Source Coding Theorem

Theorem 3.3 (Shannon’s Source Coding Theorem)

∃ a sequence $(2^{nR}, n)$ of universal source codes such that $P_e(n) \to 0$ for all source distributions $Q$ such that $H(Q) < R$.

Proof.

- Fix $R > H(Q)$ to be strictly greater than entropy.
- Define a rate for $n$ that is “fixed up” with a polynomial factor. I.e.,

$$R_n \triangleq R - |\mathcal{X}| \frac{\log(n + 1)}{n} < R$$

(21)
Source Coding Theorem

Theorem 3.3 (Shannon’s Source Coding Theorem)

∃ a sequence \((2^n R, n)\) of universal source codes such that \(P_e(n) \to 0\) for all source distributions \(Q\) such that \(H(Q) < R\).

Proof.

- Fix \(R > H(Q)\) to be strictly greater than entropy.
- Define a rate for \(n\) that is “fixed up” with a polynomial factor. I.e.,

\[
R_n \triangleq R - |X| \frac{\log(n + 1)}{n} < R
\]

(21)

- Define set of sequences that have entropy less than this rate.

\[
A_n \triangleq \{x_{1:n} \in X^n : H(P_{x_{1:n}}) \leq R_n\}
\]

(22)

\[
= \bigcup_{P \in \mathcal{P}_n} T(P) : H(P) \leq R_n
\]

(23)

...
Source Coding Theorem

... Proof of theorem 3.3 continued.

Then

\[
|A_n| = \sum_{P \in P_n} H(P) \leq R_n T(P) \leq \sum_{P \in P_n} H(P) \leq R_n 2^n H(P) \leq (n+1)|X| 2^n R_n (25)
\]

Since \(|A_n| \leq 2^n R_n\), we can index \(A_n\) with \(n R_n\) bits.

Let the encoder be:

\[
f_n(x_1:n) = \begin{cases} 
\text{lexicographic index of } x_1:n \text{ in } A_n & \text{if } x_1:n \in A_n \text{ (i.e., if } H(P_{x_1:n}) \leq R_n) \\
0 & \text{else (i.e., if } H(P_{x_1:n}) > R_n) 
\end{cases}
\]

(27)

...
Then $|A_n|$
Then $|A_n| = \sum_{P \in \mathcal{P}_n : H(P) \leq R_n} T(P)$

(26)
Source Coding Theorem

... Proof of theorem 3.3 continued.

Then

$$|A_n| = \sum_{P \in \mathcal{P}_n : H(P) \leq R_n} T(P) \leq \sum_{P \in \mathcal{P}_n : H(P) \leq R_n} 2^n H(P)$$

(24)

$$\leq (n+1)|X|2^n R_n$$

(25)

$$\leq 2^n (R_n + |X| \log(n+1))$$

(26)
Then \(|A_n| = \sum_{P \in \mathcal{P}_n: H(P) \leq R_n} T(P) \leq \sum_{P \in \mathcal{P}_n: H(P) \leq R_n} 2^{nH(P)}\) \hspace{1cm} (24)

\[ \leq \sum_{P \in \mathcal{P}_n: H(P) \leq R_n} 2^{nR_n} \]

\[ \leq 2^{nR_n} \]

\[ (26) \]
Then \( |A_n| = \sum_{P \in \mathcal{P}_n: H(P) \leq R_n} T(P) \leq \sum_{P \in \mathcal{P}_n: H(P) \leq R_n} 2^{nH(P)} \) (24)

\[
\leq \sum_{P \in \mathcal{P}_n: H(P) \leq R_n} 2^n R_n \leq (n + 1)|X|2^n R_n
\] (25)

\[
= 2^n (R_n + |X| \log(n + 1))
\] (26)
Source Coding Theorem

... Proof of theorem 3.3 continued.

Then

\[
|A_n| = \sum_{P \in \mathcal{P}_n : H(P) \leq R_n} T(P) \leq \sum_{P \in \mathcal{P}_n : H(P) \leq R_n} 2^{nH(P)}
\]  

(24)

\[
\leq \sum_{P \in \mathcal{P}_n : H(P) \leq R_n} 2^{nR_n} \leq (n + 1)|\mathcal{X}|2^{nR_n}
\]  

(25)

\[
= 2^{n(R_n + |\mathcal{X}| \log(n+1)/n)}
\]  

(26)
Source Coding Theorem

... Proof of theorem 3.3 continued.

Then

\[ |A_n| = \sum_{P \in \mathcal{P}_n : H(P) \leq R_n} T(P) \leq \sum_{P \in \mathcal{P}_n : H(P) \leq R_n} 2^{nH(P)} \]  \hspace{1cm} (24)

\[ \leq \sum_{P \in \mathcal{P}_n : H(P) \leq R_n} 2^{nR_n} \leq (n + 1)|\mathcal{X}|2^{nR_n} \]  \hspace{1cm} (25)

\[ = 2^n(R_n + |\mathcal{X}| \frac{\log(n+1)}{n}) = 2^nR \]  \hspace{1cm} (26)
Source Coding Theorem

... Proof of theorem 3.3 continued.

- Then

\[ |A_n| = \sum_{P \in \mathcal{P}_n : H(P) \leq R_n} T(P) \leq \sum_{P \in \mathcal{P}_n : H(P) \leq R_n} 2^{nH(P)} \tag{24} \]

\[ \leq \sum_{P \in \mathcal{P}_n : H(P) \leq R_n} 2^{nR_n} \leq (n + 1)|\mathcal{X}|2^{nR_n} \tag{25} \]

\[ = 2^n(R_n + |\mathcal{X}|\log(n+1)) = 2^nR \tag{26} \]

- Since \( |A_n| \leq 2^nR \), we can index \( A_n \) with \( nR \) bits.
Source Coding Theorem

... Proof of theorem 3.3 continued.

- Then \(|A_n| = \sum_{P \in \mathcal{P}_n: H(P) \leq R_n} T(P) \leq \sum_{P \in \mathcal{P}_n: H(P) \leq R_n} 2^{nH(P)} \tag{24}\)

\[
\begin{align*}
&\leq \sum_{P \in \mathcal{P}_n: H(P) \leq R_n} 2^{nR_n} \\
&\leq (n + 1)|\mathcal{X}| 2^{nR_n} \tag{25}
\end{align*}
\]

\[
\begin{align*}
&= 2^{n(R_n + |\mathcal{X}| \frac{\log(n+1)}{n})} \\
&= 2^{nR} \tag{26}
\end{align*}
\]

- Since \(|A_n| \leq 2^{nR}\), we can index \(A_n\) with \(nR\) bits.

- Let the encoder be:

\[
f_n(x_{1:n}) = \begin{cases} 
\text{lexicographic index} & \text{if } x_{1:n} \in A_n \\
0 & \text{(i.e., if } H(P_{x_{1:n}}) \leq R_n) \\
\end{cases} (\text{i.e., if } H(P_{x_{1:n}}) > R_n) \tag{27}
\]
... Proof of theorem 3.3 continued.

- Note: $f_n(\cdot)$ does not depend on the source distribution, only on the ordering and on $R_n$. 

...
Proof of theorem 3.3 continued.

- Note: $f_n(\cdot)$ does **not** depend on the source distribution, only on the ordering and on $R_n$.
- Error occurs if $x_{1:n} \notin A_n$. 

---

Source Coding Theorem
... Proof of theorem 3.3 continued.

- Note: \( f_n(\cdot) \) does not depend on the source distribution, only on the ordering and on \( R_n \).
- Error occurs if \( x_{1:n} \notin A_n \).
- We can represent this by placing types within the probability simplex, to indicate which types may be encoded. E.g., if \( |\mathcal{X}| = 3 \), then
Within the simplex, each point is potentially a type (the points with rational values with denominator $n$ and numerator between 0 and $n$).

- $H(P) < R_n$
- $H(P) > R_n$
- $H(P) = R_n$

Set of sequences that are encoded correctly.
Proof of theorem 3.3 continued.

- Within the simplex, each point is potentially a type (the points with rational values with denominator $n$ and numerator between 0 and $n$).

- Yellow region corresponds to types $P \in P_n$ whose sequences can be encoded correctly, as the rate constraint is satisfied.

\[
H(P) < R_n \\
H(P) > R_n \\
H(P) = R_n
\]
Proof of theorem 3.3 continued.

- Within the simplex, each point is potentially a type (the points with rational values with denominator $n$ and numerator between 0 and $n$).

- Yellow region corresponds to types $P \in P_n$ whose sequences can be encoded correctly, as the rate constraint is satisfied.

- Light blue corresponds to types whose sequences will result in an error.
... Proof of theorem 3.3 continued.

- We upper bound $P_e^{(n)}$ and show it $\to 0$ as $n \to \infty$ when $R > H(Q)$.
... Proof of theorem 3.3 continued.

- We upper bound $P_e^{(n)}$ and show it $\to 0$ as $n \to \infty$ when $R > H(Q)$.
- An error occurs when the sequence is not in $A^n$, thus

\begin{equation}
\sum_{P_H > R} P_e^{(n)} = 1 - Q_n(T(P_H)) \leq (n+1)|X|_{\max} P_{H > R}^n Q_n(T(P_H)) \leq (n+1)|X|^2 - n \min P_{H > R}^n D(Q||P_H)
\end{equation}

(30)
Source Coding Theorem

... Proof of theorem 3.3 continued.

- We upper bound $P_e^{(n)}$ and show it $\to 0$ as $n \to \infty$ when $R > H(Q)$.
- An error occurs when the sequence is not in $A$, thus

$$P_e^{(n)}$$

\[ (30) \]
Source Coding Theorem

... Proof of theorem 3.3 continued.

- We upper bound $P_e^{(n)}$ and show it $\rightarrow 0$ as $n \rightarrow \infty$ when $R > H(Q)$.
- An error occurs when the sequence is not in $A$, thus

$$P_e^{(n)} = 1 - Q^n(A_n)$$

(30)
Source Coding Theorem

... Proof of theorem 3.3 continued.

- We upper bound $P_e^{(n)}$ and show it → 0 as $n \to \infty$ when $R > H(Q)$.
- An error occurs when the sequence is not in $A$, thus

$$P_e^{(n)} = 1 - Q^n(Q_n) = \sum_{P: H(P) > R_n} Q^n(T(P)) \tag{28}$$

$$\leq (n + 1)|X| \max P: H(P) > R_n$$

$$R_n \uparrow R \Rightarrow R_n < R \text{ for all } n,$$

and

$$H(Q) < R_n.$$
Source Coding Theorem

... Proof of theorem 3.3 continued.

- We upper bound $P_e(n)$ and show it $\rightarrow 0$ as $n \rightarrow \infty$ when $R > H(Q)$.
- An error occurs when the sequence is not in $A$, thus

$$P_e(n) = 1 - Q^n(Q_n) = \sum_{P: H(P) > R_n} Q^n(T(P))$$

(28)

$$\leq (n + 1)|\mathcal{X}| \max_{P: H(P) > R_n} Q^n(T(P))$$

(29)

$$\leq (n + 1)|\mathcal{X}| \max_{P: H(P) > R_n} Q^n(T(P))$$

(30)

...
Source Coding Theorem

... Proof of theorem 3.3 continued.

- We upper bound $P_e^{(n)}$ and show it $\to 0$ as $n \to \infty$ when $R > H(Q)$.
- An error occurs when the sequence is not in $A$, thus

$$P_e^{(n)} = 1 - Q^n(Q_n) = \sum_{P: H(P) > R_n} Q^n(T(P)) \quad (28)$$

$$\leq (n + 1)^{|\mathcal{X}|} \max_{P: H(P) > R_n} Q^n(T(P)) \quad (29)$$

$$\leq (n + 1)^{|\mathcal{X}|} 2^{-n \left[ \min_{P: H(P) > R_n} D(P||Q) \right]} \quad (30)$$
... Proof of theorem 3.3 continued.

- We upper bound $P_e^{(n)}$ and show it $\to 0$ as $n \to \infty$ when $R > H(Q)$.
- An error occurs when the sequence is not in $A$, thus

  $$P_e^{(n)} = 1 - Q^n(Q_n) = \sum_{P: H(P) > R_n} Q^n(T(P))$$  \hspace{2cm} (28)$$

  $$\leq (n + 1)^{|X|} \max_{P: H(P) > R_n} Q^n(T(P))$$  \hspace{2cm} (29)$$

  $$\leq (n + 1)^{|X|} 2^{-n \left[ \min_{P: H(P) > R_n} D(P||Q) \right]}$$  \hspace{2cm} (30)$$

- So we have $R_n \uparrow R \Rightarrow R_n < R$ for all $n$, and $H(Q) < R$. 

...
Source Coding Theorem

... Proof of theorem 3.3 continued.

- We upper bound $P_e(n)$ and show it $\to 0$ as $n \to \infty$ when $R > H(Q)$.
- An error occurs when the sequence is not in $A$, thus

$$P_e(n) = 1 - Q^n(Q_n) = \sum_{P: H(P) > R_n} Q^n(T(P))$$

(28)

$$\leq (n + 1)^{|X|} \max_{P: H(P) > R_n} Q^n(T(P))$$

(29)

$$\leq (n + 1)^{|X|} 2^{-n \left[ \min_{P: H(P) > R_n} D(P\|Q) \right]}$$

(30)

- So we have $R_n \uparrow R \Rightarrow R_n < R$ for all $n$, and $H(Q) < R$.
- Thus, for some $n_0$, $\forall n > n_0$, we have $H(Q) < R_n$.
\[ R_{n_0-2} \quad R_{n_0-1} \quad R_{n_0} \quad R_n \]

\[ H(Q) \quad H(P) \quad R \]

- In Eq. (30) we chose \( P : H(P) > R_n \) for the current \( n \) (assuming there is one)
Source Coding Theorem

... Proof of theorem 3.3 continued.

\[ R_{n_0-2} \quad R_{n_0-1} \quad R_{n_0} \quad R_n \]

\[ H(Q) \quad H(P) \quad R \]

- In Eq. (30) we chose \( P : H(P) > R_n \) for the current \( n \) (assuming there is one)
- \( \Rightarrow H(P) > R_n > H(Q) \)
Source Coding Theorem

... Proof of theorem 3.3 continued.

\[ R_{n_0-2} \quad R_{n_0-1} \quad R_{n_0} \quad R_n \]

- In Eq. (30) we chose \( P : H(P) > R_n \) for the current \( n \) (assuming there is one)
- \( \Rightarrow H(P) > R_n > H(Q) \)
- \( \Rightarrow P \neq Q \)
In Eq. (30) we chose $P : H(P) > R_n$ for the current $n$ (assuming there is one)

$\Rightarrow H(P) > R_n > H(Q)$

$\Rightarrow P \neq Q$

$\Rightarrow D(P\|Q) > 0$ for the chosen $P$
Source Coding Theorem

... Proof of theorem 3.3 continued.

In Eq. (30) we chose $P : H(P) > R_n$ for the current $n$ (assuming there is one)

$\Rightarrow H(P) > R_n > H(Q)$

$\Rightarrow P \neq Q$

$\Rightarrow D(P∥Q) > 0$ for the chosen $P$

Thus, we get

$$P_e^{(n)} \leq (n + 1)|\mathcal{X}| 2^{-n\left[\min_{P: H(P) > R_n} D(P∥Q)\right]}$$  (31)
... Proof of theorem 3.3 continued.

In Eq. (30) we chose $P : H(P) > R_n$ for the current $n$ (assuming there is one)

$\Rightarrow H(P) > R_n > H(Q)$

$\Rightarrow P \neq Q$

$\Rightarrow D(P \| Q) > 0$ for the chosen $P$

Thus, we get

$$P_e^{(n)} \leq (n + 1)|\mathcal{X}| \cdot 2^{-n \left[ \min_{P : H(P) > R_n} D(P \| Q) \right]}$$

(31)
In Eq. (30) we chose $P : H(P) > R_n$ for the current $n$ (assuming there is one)

$\Rightarrow H(P) > R_n > H(Q)$

$\Rightarrow P \neq Q$

$\Rightarrow D(P \parallel Q) > 0$ for the chosen $P$

Thus, we get

$$P_e(n) \leq (n + 1)|\mathcal{X}| 2^{-n \left[ \min_{P : H(P) > R_n} D(P \parallel Q) \right]}$$  \hspace{1cm} (31)
Source Coding Theorem

... Proof of theorem 3.3 continued.

In Eq. (30) we chose $P : H(P) > R_n$ for the current $n$ (assuming there is one)

$\Rightarrow H(P) > R_n > H(Q)$

$\Rightarrow P \neq Q$

$\Rightarrow D(P\|Q) > 0$ for the chosen $P$

Thus, we get

$$P_e^{(n)} \leq (n + 1)|\mathcal{X}| \cdot 2^{-n \left[ \min_{P : H(P) > R_n} D(P\|Q) \right]}$$

Which implies that $P_e^{(n)} \rightarrow 0$ as $n \rightarrow \infty$. 
So far we’ve been talking about i.i.d. random variables, $X_1, X_2, \ldots$. 
So far we’ve been talking about i.i.d. random variables, $X_1, X_2, \ldots$.

In such case, each random variable has the same entropy.
Stochastic Processes

- So far we’ve been talking about i.i.d. random variables, $X_1, X_2, \ldots$.
- In such case, each random variable has the same entropy.
- When the random variables are no longer i.i.d., how can we talk about the entropy of a process?
So far we’ve been talking about i.i.d. random variables, $X_1, X_2, \ldots$.

In such case, each random variable has the same entropy.

When the random variables are no longer i.i.d., how can we talk about the entropy of a process?

We start to address that here.
Definition 4.1 ((strict-sense) Stationary stochastic Process)

A sequence of r.v.s, $X_1, X_2, \ldots, X_n$ governed by a probability distribution is strict sense stationary if it is the case that

$$p(X_{1:n} = x_{1:n}) = p(X_{1+\ell:n+\ell} = x_{1:n})$$  \hspace{1cm} (32)

for all $\ell$, for all $n$, and for all $x_{1:n} \in \mathcal{X}^n$. 
Stochastic Process

Definition 4.1 ((strict-sense) Stationary stochastic Process)

A sequence of r.v.s, $X_1, X_2, \ldots, X_n$ governed by a probability distribution is strict sense stationary if it is the case that

$$p(X_{1:n} = x_{1:n}) = p(X_{1+\ell:n+\ell} = x_{1:n})$$

(32)

for all $\ell$, for all $n$, and for all $x_{1:n} \in \mathcal{X}^n$.

Definition 4.2 (Markov process)

A stochastic process is first-order Markov if

$$p(X_{n+1} = x_{n+1} | X_{1:n} = x_{1:n}) = p(X_{n+1} = x_{n+1} | X_n = x_n)$$

(33)
Definition 4.1 ((strict-sense) Stationary stochastic Process)

A sequence of r.v.s, $X_1, X_2, \ldots, X_n$ governed by a probability distribution is strict sense stationary if it is the case that

$$p(X_{1:n} = x_{1:n}) = p(X_{1+\ell:n+\ell} = x_{1:n})$$

(32)

for all $\ell$, for all $n$, and for all $x_{1:n} \in X^n$.

Definition 4.2 (Markov process)

A stochastic process is first-order Markov if

$$p(X_{n+1} = x_{n+1} | X_{1:n} = x_{1:n}) = p(X_{n+1} = x_{n+1} | X_n = x_n)$$

(33)

In this latter case, it means that $p(x_{1:n}) = p(x_1)p(x_2|x_1) \ldots p(x_n|x_{n-1})$.
Definition 4.3 (homogeneous)

A Markov chain is time-invariant (or time-homogeneous, or just homogeneous) if \( p(x_{n+1}|x_n) \) does not depend on time. I.e., if

\[
p(X_{n+1} = b|X_n = a) = p(X_2 = b|X_1 = a) \quad \forall a, b, n
\] (34)
Definition 4.3 (homogeneous)

A Markov chain is time-invariant (or time-homogeneous, or just homogeneous) if \( p(x_{n+1} \mid x_n) \) does not depend on time. I.e., if

\[
p(X_{n+1} = b \mid X_n = a) = p(X_2 = b \mid X_1 = a) \quad \forall a, b, nanumber{(34)}
\]

In such case, there is a fixed transition matrix \( P = [p_{ij}]_{ij} \) with \( p_{ij} = p(X_{n+1} = j \mid X_n = i) \) that can be drawn as a directed graph with arrows pointing between states that have non-zero transition.
Stochastic Process

Definition 4.4 (irreducible)

A Markov chain is **irreducible** if $p_{ij}(n) > 0$ for all $i, j$ and for some $n$ where $p_{ij}(n) = p(X_n = j|X_0 = i)$.

This is if it is possible to get from all states to all others with non-zero probability.
A Markov chain is irreducible if $p_{ij}(n) > 0$ for all $i, j$ and for some $n$ where $p_{ij}(n) = p(X_n = j | X_0 = i)$.

This is if it is possible to get from all states to all others with non-zero probability.

- Also recall note, matrix-vector for state probability at time $n + 1$ given that at time $n$.

$$p(x_{n+1}) = \sum_{x_n} p(x_n)p_{x_n,x_{n+1}}$$  \hspace{1cm} (35)
Definition 4.4 (irreducible)

A Markov chain is irreducible if \( p_{ij}(n) > 0 \) for all \( i, j \) and for some \( n \) where \( p_{ij}(n) = p(X_n = j | X_0 = i) \).

This is if it is possible to get from all states to all others with non-zero probability.

- Also recall note, matrix-vector for state probability at time \( n + 1 \) given that at time \( n \).

\[
p(x_{n+1}) = \sum_{x_n} p(x_n) p_{x_n, x_{n+1}}
\]  

(first order) Markov chain is stationary if \( p(x_{n+1}) = p(x_n) \)
Definition 4.5

A Markov chain is periodic if \( d(i) > 1 \) with

\[
d(i) = \gcd\{n : p_{ii}(n) > 0\}
\]

Note that this is the gcd of the epochs at which return to the same state is possible.
Stochastic Process

Example:

\[ P = \begin{pmatrix} 1 - \alpha & \alpha \\ \beta & 1 - \beta \end{pmatrix} \]
Example:

\[ P = \begin{pmatrix} 1 - \alpha & \alpha \\ \beta & 1 - \beta \end{pmatrix} \]

If \( \mu = [p_1 p_2]^T \) is stationary distribution then we must have that \( \mu^T P = \mu^T \).
Example:

\[ P = \begin{pmatrix} 1 - \alpha & \alpha \\ \beta & 1 - \beta \end{pmatrix} \]

If \( \mu = [p_1 p_2]^T \) is stationary distribution then we must have that \( \mu^T P = \mu^T \).

In fact, in this case \( \mu = \left[ \frac{\beta}{\alpha+\beta}, \frac{\alpha}{\alpha+\beta} \right] \).
Stochastic Process

- Example:

\[ P = \begin{pmatrix} 1 - \alpha & \alpha \\ \beta & 1 - \beta \end{pmatrix} \]

- If \( \mu = [p_1 \, p_2]^T \) is stationary distribution then we must have that \( \mu^T P = \mu^T \).

- In fact, in this case \( \mu = \left[ \frac{\beta}{\alpha + \beta}, \frac{\alpha}{\alpha + \beta} \right] \).

- More facts about Markov chains and stochastic processes: Great source is the text: see “Probability and Random Processes”, Grimmett and Stirzaker.
Entropy rates

- Stochastic processes have entropy rates, which intuitively is the amount of new information, on average, that is provided by the stochastic process at each time step.
Entropy rates

- Stochastic processes have entropy rates, which intuitively is the amount of new information, on average, that is provided by the stochastic process at each time step.
- More formally
Entropy rates

- Stochastic processes have entropy rates, which intuitively is the amount of new information, on average, that is provided by the stochastic process at each time step.

- More formally

**Definition 5.1**

The **entropy rate of a stochastic process** \( \{X_i\}_i \) is defined as

\[
H(\mathcal{X}) \triangleq \lim_{n \to \infty} \frac{1}{n} H(X_1, X_2, \ldots, X_n)
\]

when it exists.
Entropy rates

- Stochastic processes have entropy rates, which intuitively is the amount of new information, on average, that is provided by the stochastic process at each time step.

- More formally

**Definition 5.1**

The entropy rate of a stochastic process \( \{X_i\}_i \) is defined as

\[
H(\mathcal{X}) \triangleq \lim_{n \to \infty} \frac{1}{n} H(X_1, X_2, \ldots, X_n)
\]

(37)

when it exists.

- So, as can be seen, it is the per symbol entropy given by the stochastic process when \( n \) gets large.
Examples

- i.i.d. set of r.v.s all $\sim p(x)$ then

$$H(X) = \lim_{n \to \infty} \frac{H(X_{x_1:n})}{n} = \frac{\sum_{i=1}^{n} H(X_i)}{n} = H(X_1) \quad (38)$$
Examples

- **I.i.d. set of r.v.s all \( \sim p(x) \) then**

\[
H(\mathcal{X}) = \lim_{n \to \infty} \frac{H(X_{x1:n})}{n} = \frac{\sum_{i=1}^{n} H(X_i)}{n} = H(X_1) \tag{38}
\]

- **Independent but not identically distributed:**

\[
\lim_{n \to \infty} \frac{\sum_{i=1}^{n} H(X_i)}{n} = ? \tag{39}
\]

in this case it might not exist.
Examples

- l.i.d. set of r.v.s all $\sim p(x)$ then
  
  $$H(X) = \lim_{n \to \infty} \frac{H(X_{x_1\ldots x_n})}{n} = \sum_{i=1}^{n} \frac{H(X_i)}{n} = H(X_1)$$
  \hspace{1cm} (38)

- Independent but not identically distributed:
  
  $$\lim_{n \to \infty} \frac{\sum_{i=1}^{n} H(X_i)}{n} = ?$$
  \hspace{1cm} (39)
  
  in this case it might not exist.

- Example when it doesn’t exist. Let $p_i = P(X_i = 1)$. Define it as
  
  $$p_i = \begin{cases} 
  0.5 & \text{if } 2k < \log \log i \leq 2k + 1 \\
  0 & \text{if } 2k + 1 < \log \log i \leq 2k + 2 
  \end{cases}$$
  \hspace{1cm} (40)
Definition 5.2

Again, assume a stochastic process and define the following rate:

\[ H'(\mathcal{X}) \triangleq \lim_{n \to \infty} H(X_n | X_{n-1}, X_{n-2}, \ldots, X_1) \]  

(41)

Theorem 5.3

For stationary stochastic process, \( H(X_n | X_{n-1}, X_{n-2}, \ldots, X_1) \) is decreasing in \( n \) and has a limit, lets call it \( H'(\mathcal{X}) \).

Proof.

\[ H(X_{n+1} | X_1, \ldots, X_n) \leq H(X_{n+1} | X_2, \ldots, X_n) = H(X_1 | X_1, \ldots, X_n) \]  

(42)

since we have a decreasing sequence with lower bound 0, it has a limit \( H'(\mathcal{X}) \).
Alternative Definition

Definition 5.2
Again, assume a stochastic process and define the following rate:

\[ H'(\mathcal{X}) \triangleq \lim_{n \to \infty} H(X_n|X_{n-1}, X_{n-2}, \ldots, X_1) \] (41)

Theorem 5.3
For stationary stochastic process, \( H(X_n|X_{n-1}, X_{n-2}, \ldots, X_1) \) is decreasing in \( n \) and has a limit, lets call it \( H'(\mathcal{X}) \).
Alternative Definition

Definition 5.2

Again, assume a stochastic process and define the following rate:

\[ H'(\mathcal{X}) \triangleq \lim_{n \to \infty} H(X_n|X_{n-1}, X_{n_2}, \ldots, X_n) \]  

(41)

Theorem 5.3

For stationary stochastic process, \( H(X_n|X_{n-1}, X_{n_2}, \ldots, X_n) \) is decreasing in \( n \) and has a limit, lets call it \( H'(\mathcal{X}) \).

Proof.

\[ H(X_{n+1}|X_1, \ldots, X_n) \leq H(X_{n+1}|X_2, \ldots, X_n) = H(X_{n+1}|X_1, \ldots, X_n) \]

(42)
Alternative Definition

Definition 5.2

Again, assume a stochastic process and define the following rate:

$$H'(\mathcal{X}) \triangleq \lim_{n \to \infty} H(X_n|X_{n-1}, X_{n_2}, \ldots, X_n)$$  \hspace{1cm} (41)

Theorem 5.3

For stationary stochastic process, $H(X_n|X_{n-1}, X_{n_2}, \ldots, X_n)$ is decreasing in $n$ and has a limit, lets call it $H'(\mathcal{X})$.

Proof.

$$H(X_{n+1}|X_1, \ldots, X_n) \leq H(X_{n+1}|X_2, \ldots, X_n) = H(X_1|X_1, \ldots, X_n)$$  \hspace{1cm} (42)

since we have a decreasing sequence with lower bound 0, it has a limit $H'$. 

Entropy rates or entropy rate

- Cesáro mean: if $a_n \to a$ and $b_n = \frac{1}{n} \sum_{i=1}^{n} a_n$ then $b_n \to a$
Entropy rates or entropy rate

- Cesáro mean: if $a_n \to a$ and $b_n = \frac{1}{n} \sum_{i=1}^{n} a_n$ then $b_n \to a$
- Key idea is that most of the terms in the sum are close to $a$, so the average is also close to $a$ (formal proof in book).
Entropy rates or entropy rate

- Cesáro mean: if $a_n \to a$ and $b_n = \frac{1}{n} \sum_{i=1}^{n} a_n$ then $b_n \to a$
- Key idea is that most of the terms in the sum are close to $a$, so the average is also close to $a$ (formal proof in book).
- This then gives:
Entropy rates or entropy rate

- Cesáro mean: if \( a_n \to a \) and \( b_n = \frac{1}{n} \sum_{i=1}^{n} a_n \) then \( b_n \to a \)
- Key idea is that most of the terms in the sum are close to \( a \), so the average is also close to \( a \) (formal proof in book).
- This then gives:

**Theorem 5.4**

*We have that for stationary stochastic processes*

\[
H'(X) = H(X)
\]  
(43)
Entropy rates or entropy rate

- Cesáro mean: if $a_n \to a$ and $b_n = \frac{1}{n} \sum_{i=1}^{n} a_n$ then $b_n \to a$
- Key idea is that most of the terms in the sum are close to $a$, so the average is also close to $a$ (formal proof in book).
- This then gives:

**Theorem 5.4**

*We have that for stationary stochastic processes*

$$H'(\mathcal{X}) = H(\mathcal{X})$$  \hspace{1cm} (43)
Entropy rates or entropy rate

- Cesáro mean: if \( a_n \to a \) and \( b_n = \frac{1}{n} \sum_{i=1}^{n} a_n \) then \( b_n \to a \)
- Key idea is that most of the terms in the sum are close to \( a \), so the average is also close to \( a \) (formal proof in book).
- This then gives:

**Theorem 5.4**

We have that for stationary stochastic processes

\[
H'(\mathcal{X}) = H(\mathcal{X})
\]  

(43)

**Proof.**

\[
b_n = \frac{H(X_1, X_2, \ldots, X_n)}{n} = \frac{1}{n} \sum_{i=1}^{n} H(X_i | X_{i-1}, \ldots, X_1)
\]

(44)

and \( a_n \to H'(\mathcal{X}) \) so \( b_n \to H'(\mathcal{X}) \) but by definition \( b_n \to H(\mathcal{X}) \)
Entropy rates or entropy rate

- Cesàro mean: if \( a_n \to a \) and \( b_n = \frac{1}{n} \sum_{i=1}^{n} a_i \) then \( b_n \to a \)
- Key idea is that most of the terms in the sum are close to \( a \), so the average is also close to \( a \) (formal proof in book).
- This then gives:

**Theorem 5.4**

*We have that for stationary stochastic processes*

\[
H'(X) = H(X)
\]  

(43)

**Proof.**

\[
b_n = \frac{H(X_1, X_2, \ldots, X_n)}{n} = \frac{1}{n} \sum_{i=1}^{n} H(X_i | X_{i-1}, \ldots, X_1) = a_n
\]  

(44)

and \( a_n \to H'(X) \) so \( b_n \to H'(X) \) but by definition \( b_n \to H(X) \)
Note that for any stationary ergodic (loosely, time and ensemble averages are the same) process, we have

\[-\frac{1}{n} \log p(x_1, \ldots, x_n) \rightarrow H(X)\]  

(45)
**Entropy rate**

- Note that for any stationary ergodic (loosely, time and ensemble averages are the same) process, we have
  
  \[
  -\frac{1}{n} \log p(x_1, \ldots, x_n) \rightarrow H(X) \quad (45)
  \]

- With this, we can prove AEP-like theorems and prove the source coding theorem for such processes, but we need more machinery to do so. This is done in section 16.8 Shannon-McMillan-Breiman Theorem (General AEP) (page 644) in our book.
Entropy rate and stationary Markov chain

When the process is a stationary Markov chain, entropy rate has a nice form.
Entropy rate and stationary Markov chain

- When the process is a stationary Markov chain, entropy rate has a nice form.

That is

\[
H(X) = H'(X) = \lim_{n \to \infty} H(X_n|X_{n-1}, \ldots, X_1) = \lim_{n \to \infty} H(X_n|X_{n-1}) = H(X_2|X_1)
\]

\[
= - \sum_{ij} p(x_2, x_1) \log p(x_2|x_1) = \sum_i \mu_i \left[ - \sum_j p_{ij} \log p_{ij} \right]
\]

where again \( \mu \) is the stationary distribution and \( p_{ij} \) is the transition probability from \( i \) to \( j \).
Entropy rate and stationary Markov chain

- When the process is a stationary Markov chain, entropy rate has a nice form.

That is

\[ H(X) = H'(X) = \lim_{n \to \infty} H(X_n|X_{n-1}, \ldots, X_1) \]

\[ = \lim_{n \to \infty} H(X_n|X_{n-1}) = H(X_2|X_1) \]

\[ = - \sum_{ij} p(x_2, x_1) \log p(x_2|x_1) = \sum_i \mu_i \left[ - \sum_j p_{ij} \log p_{ij} \right] \]

(48)

where again \( \mu \) is the stationary distribution and \( p_{ij} \) is the transition probability from \( i \) to \( j \).

- Ex: previous figure

\[ H(X) = H(X_2|X_1) = \frac{\beta}{\alpha+\beta} H(\alpha) + \frac{\alpha}{\alpha+\beta} H(\beta). \]
Ex: random walk on weighted undirected graph

- Assume irreducible and aperiodic so unique stationary distribution.
Ex: random walk on weighted undirected graph

- Assume irreducible and aperiodic so unique stationary distribution.
- Graph $G = (V, E)$ with $m$ nodes labeled $\{1, 2, \ldots, m\}$ and edges with weight $w_{ij} \geq 0$. 

\begin{align*}
\text{Random walk: start at a node, say } i, \text{ and choose next node with probability proportional to edge weight, i.e., } p_{ij} &= \frac{w_{ij}}{\sum_j w_{ij}}.
\end{align*}
Ex: random walk on weighted undirected graph

- Assume irreducible and aperiodic so unique stationary distribution.
- Graph $G = (V, E)$ with $m$ nodes labeled $\{1, 2, \ldots, m\}$ and edges with weight $w_{ij} \geq 0$.
- Random walk: start at a node, say $i$, and choose next node with probability proportional to edge weight, i.e., $p_{ij}$ as

$$p_{ij} = \frac{w_{ij}}{\sum_j w_{ij}} = \frac{w_{ij}}{w_i}$$

(49)

where $w_i \triangleq \sum_j w_{ij}$.
Ex: random walk on weighted undirected graph

- Assume irreducible and aperiodic so unique stationary distribution.
- Graph $G = (V, E)$ with $m$ nodes labeled $\{1, 2, \ldots, m\}$ and edges with weight $w_{ij} \geq 0$.
- Random walk: start at a node, say $i$, and choose next node with probability proportional to edge weight, i.e., $p_{ij}$ as
  \[
  p_{ij} = \frac{w_{ij}}{\sum_j w_{ij}} = \frac{w_{ij}}{w_i}
  \]
  (49)

  where $w_i \triangleq \sum_j w_{ij}$.
- Guess that stationary distribution has probability proportional to $w_i$. 
Ex: random walk on weighted undirected graph

- Assume irreducible and aperiodic so unique stationary distribution.
- Graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with $m$ nodes labeled $\{1, 2, \ldots, m\}$ and edges with weight $w_{ij} \geq 0$.
- Random walk: start at a node, say $i$, and choose next node with probability proportional to edge weight, i.e., $p_{ij}$ as

$$p_{ij} = \frac{w_{ij}}{\sum_j w_{ij}} = \frac{w_{ij}}{w_i} \quad (49)$$

where $w_i \triangleq \sum_j w_{ij}$.

- Guess that stationary distribution has probability proportional to $w_i$.
- If $w = \sum_{i,j:j>i} w_{ij}$ then $\sum_i w_i = 2w$, so guess as stationary distribution $\mu$ with $\mu_i = \frac{w_i}{2w}$. 

Prof. Jeff Bilmes
Ex: random walk on weighted undirected graph

• This is stationary since

\[ \forall j, \mu'_j = \sum_i \mu_i P_{ij} = \sum_i \frac{w_i w_{ij}}{2w} \quad (50) \]

\[ = \sum_i \frac{1}{2w} w_{ij} = \frac{w_j}{2w} = \mu_j \quad (51) \]
Ex: random walk on weighted undirected graph

- This is stationary since

\[
\forall j, \mu'_j = \sum_i \mu_i P_{ij} = \sum_i \frac{w_i}{2w} \frac{w_{ij}}{w_i} = \sum_i \frac{1}{2w} w_{ij} = \frac{w_j}{2w} = \mu_j
\]

(50)

(51)

- Can swap edges elsewhere (i.e., edges between nodes not including \(i\)), does not change the stationary condition which is local.
Ex: random walk on weighted undirected graph

This is stationary since

$$\forall j, \mu'_j = \sum_i \mu_i P_{ij} = \sum_i \frac{w_i w_{ij}}{2w} = \mu_j$$  \hspace{1cm} (50)

$$= \sum_i \frac{1}{2w} w_{ij} = \frac{w_j}{2w} = \mu_j$$  \hspace{1cm} (51)

Can swap edges elsewhere (i.e., edges between nodes not including \(i\)), does not change the stationary condition which is local.

Note chain is aperiodic since \(w_{ii} = 0\). This is because

$$2w = \sum_i w_i = \sum_{ij} w_{ij} = \sum_{ij:i=j} w_{ij} + \sum_{ij:j>i} w_{ij} + \sum_{ij:j<i} w_{ij}$$  \hspace{1cm} (52)

$$= \sum_{ij:i=j} w_{ij} + w + \sum_{ij:j<i} w_{ij}$$  \hspace{1cm} (53)

and $$2w - w = \sum_{ij:i=j} w_{ij} + \sum_{ij:j<i} w_{ij} \Rightarrow \sum_{ij:i=j} w_{ij} = 0 \Rightarrow w_{ii} = 0.$$
What is entropy of this random walk

\[ H(X) = H(X_2 | X_1) = - \sum_{i} \mu_i \sum_{j} p_{ij} \log p_{ij} \]  

(54)

\[ = - \sum_{i} \frac{w_i}{2w} \sum_{j} \frac{w_{ij}}{w_i} \log \frac{w_{ij}}{w_i} = - \sum_{ij} \frac{w_{ij}}{2w} \log \frac{w_{ij}}{w_i} \]  

(55)

\[ = - \sum_{ij} \frac{w_{ij}}{2w} \log \left[ \frac{w_{ij}}{2w} \frac{2w}{w_i} \right] \]  

(56)

\[ = - \sum_{ij} \frac{w_{ij}}{2w} \log \frac{w_{ij}}{2w} + \sum_{ij} \frac{w_{ij}}{2w} \log \frac{w_i}{2w} \]  

(57)

\[ = H(\ldots, \frac{w_{ij}}{2w}, \ldots) - H(\ldots, \frac{w_i}{2w}, \ldots) \]  

(58)

\[ = H(\text{overall edge uncertainty}) \]  

(59)

\[ - H(\text{overall node uncertainty, stat. cond}) \]  

(60)
An HMM is a distribution \( p(X_{1:n}, Y_{1:n}) \) over \( 2n \) random variables that factors in a particular way.
An HMM is a distribution $p(X_{1:n}, Y_{1:n})$ over $2n$ random variables that factors in a particular way.

Easiest way to depict all of the factorization properties is to use a graphical model, as in the below, where $n = 5$:
Hidden Markov models (HMMs)

- An HMM is a distribution $p(X_{1:n}, Y_{1:n})$ over $2n$ random variables that factors in a particular way.
- Easiest way to depict all of the factorization properties is to use a graphical model, as in the below, where $n = 5$:

Let $X_1, X_2, \ldots, X_n$ be a stationary Markov chain.
Hidden Markov models (HMMs)

- An HMM is a distribution $p(X_{1:n}, Y_{1:n})$ over $2n$ random variables that factors in a particular way.
- Easiest way to depict all of the factorization properties is to use a graphical model, as in the below, where $n = 5$:

Let $X_1, X_2, \ldots, X_n$ be a stationary Markov chain.

Let $Y_{1:n}$ be a random function of this Markov chain. I.e.,

\[
Y_i = \begin{cases} 
\phi_1(X_i) & \text{with probability } p_1 \\
\phi_2(X_i) & \text{with probability } p_2 \\
& \vdots \\
\phi_m(X_i) & \text{with probability } p_m 
\end{cases} = \phi_N(X_i) \quad (61)
\]

where $N \in \{1, 2, \ldots, m\}$ itself is a random variable.
Note that the stochastic process $Y_1, Y_2, \ldots$ does not form a Markov chain in general. Why?
HMMs

Note that the stochastic process $Y_1, Y_2, \ldots$ does not form a Markov chain in general. Why? Because it does not satisfy the first order Markov assumption, nor any order Markov assumption in general.
HMMs

- Note that the stochastic process $Y_1, Y_2 \ldots$ does not form a Markov chain in general. Why? because it does not satisfy the first order Markov assumption, nor any order Markov assumption in general.

- If $\{X_i\}_i$ is stationary, then is $\{Y_i\}_i$ a stationary stochastic process?
Note that the stochastic process $Y_1, Y_2, \ldots$ does not form a Markov chain in general. Why? because it does not satisfy the first order Markov assumption, nor any order Markov assumption in general.

If $\{X_i\}_i$ is stationary, then is $\{Y_i\}_i$ a stationary stochastic process? Yes. Possible HW problem, so no more given here.
HMMs

- Note that the stochastic process $Y_1, Y_2, \ldots$ does not form a Markov chain in general. Why? because it does not satisfy the first order Markov assumption, nor any order Markov assumption in general.

- If $\{X_i\}_i$ is stationary, then is $\{Y_i\}_i$ a stationary stochastic process? Yes. Possible HW problem, so no more given here.

- We can compute the entropy rate of $\{Y_i\}_i$, i.e.,
  
  $H(\mathcal{Y}) = H \lim_{n \to \infty} (Y_n | Y_{n-1}, \ldots, Y_1)$

  but it is ugly, so instead we compute upper and lower bounds.
Note that the stochastic process $Y_1, Y_2, \ldots$ does not form a Markov chain in general. Why? because it does not satisfy the first order Markov assumption, nor any order Markov assumption in general.

If $\{X_i\}_i$ is stationary, then is $\{Y_i\}_i$ a stationary stochastic process? Yes. Possible HW problem, so no more given here.

We can compute the entropy rate of $\{Y_i\}_i$, i.e.,

$$H(Y) = H \lim_{n \to \infty} (Y_n|Y_{n-1}, \ldots, Y_1)$$

but it is ugly, so instead we compute upper and lower bounds.

Upper bound(s):

$$H(Y_n|Y_{n-1}, \ldots, Y_1) = H(Y_{n+1}|Y_n, \ldots, Y_2) \geq H(Y_{n+1}|Y_1, \ldots, Y_n) \quad (62)$$

$$\geq H(Y_{n+2}|Y_{n+1}, \ldots, Y_1) \geq \cdots \geq H(Y) \quad (63)$$
A lower bound is given by $H(Y_n|Y_{n-1}, \ldots, Y_2, X_1) \leq H(Y)$ because

\begin{align*}
H(Y_n|Y_{n-1}, \ldots, Y_2, X_1) &= H(Y_n|Y_{n-1}, \ldots, Y_2, Y_1, X_1) \\ &= H(Y_n|Y_{n-1}, \ldots, Y_1, X_1, X_0, X_{-1}, \ldots, X_{-k}) \\ &= H(Y_n|Y_{n-1}, \ldots, Y_1, X_1, X_0, X_{-1}, \ldots, X_{-k}, Y_0, \ldots, Y_{-k}) \\ &\leq H(Y_n|Y_{n-1}, \ldots, Y_1, Y_0, \ldots, Y_{-k}) \\ &= H(Y_{n+k+1}|Y_{n+k}, \ldots, Y_1)
\end{align*}
A lower bound is given by $H(Y_n|Y_{n-1}, \ldots, Y_2, X_1) \leq H(Y)$ because

$$H(Y_n|Y_{n-1}, \ldots, Y_2, X_1) = H(Y_n|Y_{n-1}, \ldots, Y_2, Y_1, X_1)$$  \hspace{1cm} (64)

$$= H(Y_n|Y_{n-1}, \ldots, Y_1, X_1, X_0, X_{-1}, \ldots, X_{-k})$$ \hspace{1cm} (65)

$$= H(Y_n|Y_{n-1}, \ldots, Y_1, X_1, X_0, X_{-1}, \ldots, X_{-k}, Y_0, \ldots, Y_{-k})$$

$$\leq H(Y_n|Y_{n-1}, \ldots, Y_1, Y_0, \ldots, Y_{-k})$$ \hspace{1cm} (66)

$$= H(Y_{n+k+1}|Y_{n+k}, \ldots, Y_1)$$ \hspace{1cm} (67)

So summarizing the bounds on the HMM information rates, we have

$$H(Y_n|Y_{n-1}, \ldots, Y_1, X) \leq H(Y) \leq H(Y_n|Y_{n-1}, \ldots, Y_1)$$ \hspace{1cm} (68)
Lemma 5.5 (ever shrinking sandwich)

\[ H(Y_n|Y_{n-1}, \ldots, Y_1) - H(Y_n|Y_{n-1}, \ldots, Y_1, X) \rightarrow 0 \] (69)

Proof.

\[
\begin{align*}
H(Y_n|Y_{n-1}, \ldots, Y_1) - H(Y_n|Y_{n-1}, \ldots, Y_1, X) &= I(Y_n; X_1|Y_{n-1}, \ldots, Y_1) \\
&\leq H(X) = H(X_1)
\end{align*}
\] (70)
Lemma 5.5 (ever shrinking sandwich)

\[ H(Y_n|Y_{n-1}, \ldots, Y_1) - H(Y_n|Y_{n-1}, \ldots, Y_1, X) \to 0 \]  

(69)

Proof.

\[ H(Y_n|Y_{n-1}, \ldots, Y_1) - H(Y_n|Y_{n-1}, \ldots, Y_1, X) = I(Y_n; X_1|Y_{n-1}, \ldots, Y_1) \]

\[ \leq H(X) = H(X_1) \]  

(70)

Now,  

\[ \lim_{n \to \infty} I(X; Y_1, \ldots, Y_n) = \lim_{n \to \infty} \sum_{i=1}^{n} I(X_1; Y_i|Y_{1:i-1}) \]  

(71)

\[ = \sum_{i=1}^{\infty} I(X_1; Y_i|Y_{1:i-1}) \leq H(X) < \infty \]  

(72)
Lemma 5.5 (ever shrinking sandwich)

\[ H(Y_n|Y_{n-1}, \ldots, Y_1) - H(Y_n|Y_{n-1}, \ldots, Y_1, X) \to 0 \]  

(69)

Proof.

\[ H(Y_n|Y_{n-1}, \ldots, Y_1) - H(Y_n|Y_{n-1}, \ldots, Y_1, X) = I(Y_n; X_1|Y_{n-1}, \ldots, Y_1) \leq H(X) = H(X_1) \]  

(70)

Now,

\[ \lim_{n \to \infty} I(X; Y_1, \ldots, Y_n) = \lim_{n \to \infty} \sum_{i=1}^{n} I(X_1; Y_i|Y_1:i-1) \]

(71)

\[ = \sum_{i=1}^{\infty} I(X_1; Y_i|Y_1:i-1) \leq H(X) < \infty \]  

(72)

So an infinite sum is constant, must mean the terms \( \to 0 \) as \( n \to \infty \). Thus, each of the terms \( I(X_1; Y_i|Y_1:i-1) \to 0 \) as \( n \to \infty \). \( \square \)
Summarizing, we have

$$\lim_{n \to \infty} H(Y_n|Y_{n-1}, \ldots, Y_1, X) = H(Y) = \lim_{n \to \infty} H(Y_n|Y_{n-1}, \ldots, Y_1)$$

(73)