- The channel capacity C for an additive white Gaussian noise (AWGN) channel:

$$C = W \log_2\left(1 + \frac{\overline{P}}{N_0 W}\right) \quad \text{bits per second}$$

- Spectral efficiency:

$$\frac{C}{W} \ \text{ bits/s/Hz} = \log\left(1 + \frac{\overline{P}}{N_0 W}\right)$$

- Energy per bit:

$$E_b = \frac{\overline{P}\cancel{T}\ \text{J}}{\cancel{T}W \log\left(1 + \frac{\overline{P}}{N_0 W}\right)}$$

Energy per bit relative to noise level:

$$\frac{E_b}{N_0} = \left(\frac{\overline{P}/N_0 W}{\log\left(1 + \overline{P}/N_0 W\right)}\right)_{\text{when minimized}}$$

- Capacity for a parallel gaussian channel / vector gaussian channel:

$$C = \sum_{j=1}^{d} \frac{1}{2} \log_2\left(1 + \frac{P_j^*}{\sigma_j^2}\right)$$

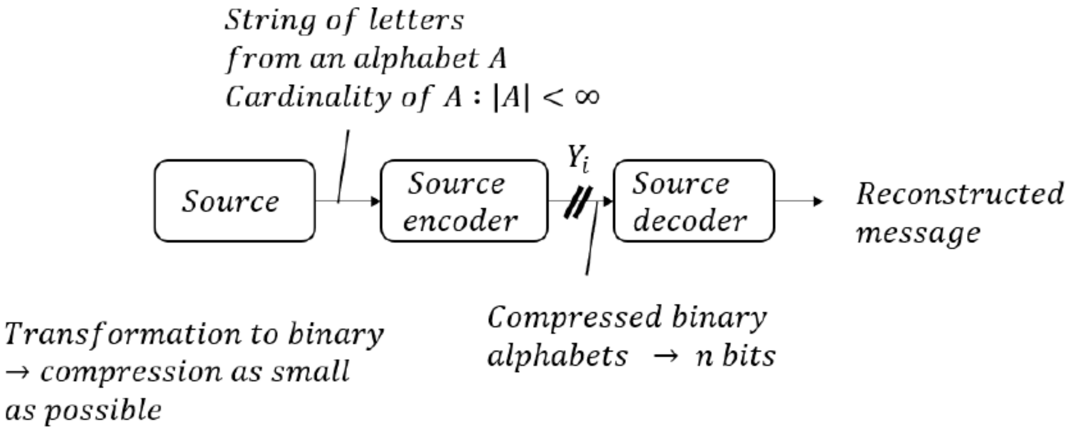
$$P_j^* = \left[\lambda - \sigma_j^2\right]_+, \ \text{where } j = 1, 2, 3, \ldots, d.$$

(20250102#1)

What is information theory about?

Information theory studies the limits of communication.

Why study? → minimize storage, transmission requirement



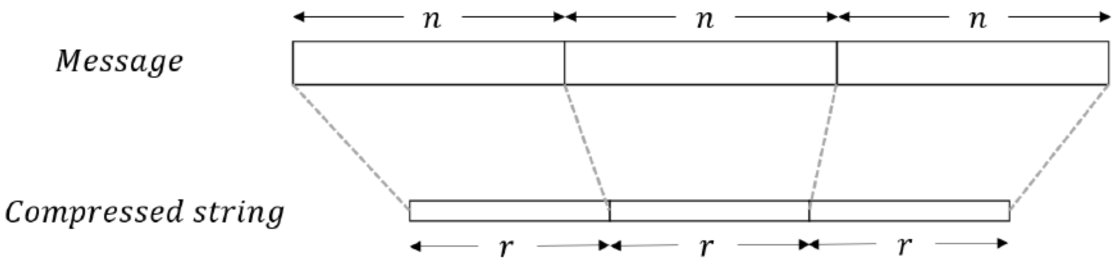
(20250102#2)

What is block source coding? How is it different from other coding schemes?

Zip compressor etc → variable length compression (variable length source coding)

We look at something simpler (for now) → fixed length coding or block source coding.

Block codes



Definition: An $(n, r)-$ binary block code is a pair of mappings

$$f_n : \ A^n \to \{0,1\}^r \quad \text{encoder}$$

$$\phi_n: \ \{0,1\}^r \to A^n \quad \text{decoder}$$

(20250102#3)

If we can't have any errors in block source coding, what conditions must be satisifed?

Can't have any errors (f_n and ϕ_n should inverse of each other) $\implies$ f_n and ϕ_n are one-to-one.

$$2^r \geq |A|^n$$

where r is the number of bits to which string of length n is compressed to.

$$r \log_2 2 \geq n \log_2 |A|$$

$$\implies \frac{r}{n} \geq \log_2 |A| \text{ bits per symbol}$$

(20250102#4)

Error probability expression for block source coding:

Let the source emit symbols $X^n = (X_1, X_2, \ldots, X_n)$ Then, error probability

$$e(f_n, \phi_n) = \Pr\{\phi_n(f_n(X_1, \ldots, X_n)) \neq (X_1, \ldots, X_n)\}$$

We would like to keep $e(f_n, \phi_n) \leq \epsilon$ (with high probability) and at the same time keep r to be as small as possible.

(20250102#5)

Define $r(n, \epsilon)$:

It is the smallest r for which we have an (n, r) binary block code with error $e(f_n, \phi_n) \leq \epsilon$.

(20250102#6)

State Shannon's source coding theorem (1948):

$X_1, X_2, \ldots$ are independent and identically distributed with pmf P_X on the alphabet A. Fix $0 < \epsilon < 1$. Then,

$$\lim_{n\to\infty} \frac{r(n,\epsilon)}{n} = H(P_X) = \sum_{a\in A} P_X(a) \log \frac{1}{P_X(a)} = H(X) \rightarrow \text{ entropy}$$

(20250102#7)

Define discrete memoryless source:

Definition: Discrete Memoryless Source (DMS)

A source is said to be a *Discrete Memoryless Source* if it emits symbols from a finite alphabet $\mathcal{X}$, and for any $n \geq 1$, the joint probability distribution of the output sequence $(X_1, X_2, \ldots, X_n)$ satisfies:

$$P_{X_1,X_2,\ldots,X_n}(x_1, x_2, \ldots, x_n) = \prod_{i=1}^{n} P_X(x_i),$$

where P_X is a fixed probability distribution over $\mathcal{X}$.

This means that each symbol X_i is independently and identically distributed (i.i.d.) according to P_X, and the output at each time is independent of past outputs.

(20250102#8)

What are some important observations regarding Shannon's source coding theorem (1948)?

---

1. The limit doesn't depend on ϵ so long as $0 < \epsilon < 1$.
2. $r(n,\epsilon) = nH(P_X) + \text{correction term } \; o(n)$, where this correction term grows sublinearly to n, meaning $o(n)/n \to 0$ as $n \to \infty$.
3. Think of $A' = A \cup \phi$ where ϕ is the null alphabet.

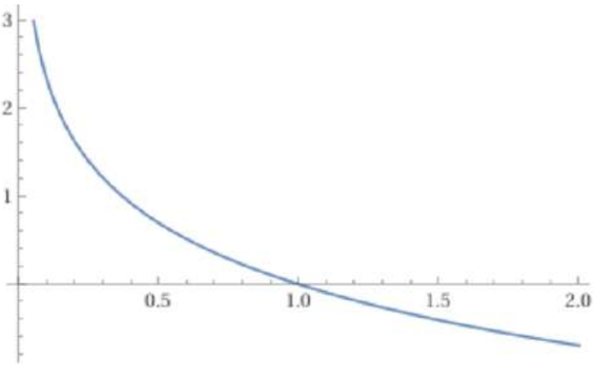
$$H(P_X) = \sum_{a\in A'} P_X(a) \log \frac{1}{P_X(a)}$$

where there will be one extra term appearing $0\log(1/0)$. We can take $0\log(1/0) = 0$ as non-sensical symbols has no probability of appearing in the possible set of strings and hence it is not going to change $H(P_X)$.

4. Swapping letters only changes the semantics, it doesn't change compression as $H(P_X)$ is invariant to permutation of the letters.

5. "Entropy"

   Plot of $log(1/x)$ vs x



- The function $\log\left(\frac{1}{t}\right) = -\log t$ is a convex function over the interval $(0, 1]$, and it diverges to ∞ as $t \to 0^+$.
- The plot of $\log(1/t)$ vs. t illustrates the contribution of events with different probabilities to the entropy:
  - As $t \to 0$, $\log(1/t) \to \infty$: rare events contribute significantly to the entropy.
  - As $t \to 1$, $\log(1/t) \to 0$: highly probable events contribute little to the entropy.
- The convexity of $\log(1/t)$ implies that entropy is a concave function of the probability distribution P. This is foundational in proving the concavity of entropy:

$$H(\lambda P_1 + (1-\lambda)P_2) \geq \lambda H(P_1) + (1-\lambda)H(P_2), \quad \forall \lambda \in [0, 1]$$

- From a geometric perspective, the shape of the $\log(1/t)$ curve shows that shifting probability mass from likely outcomes to unlikely ones increases entropy, which aligns with the intuition that a more uncertain distribution carries more information.

(20250102#9)

Prove that entropy is a concave function:

- Let $\mathcal{X}$ be a finite set, and let $\mathcal{P}(\mathcal{X})$ denote the probability simplex over $\mathcal{X}$, i.e., the set of all probability distributions on $\mathcal{X}$.

- Define the Shannon entropy function $H : \mathcal{P}(\mathcal{X}) \to \mathbb{R}$ as:

$$H(P) = -\sum_{x\in\mathcal{X}} P(x)\log P(x)$$

where we adopt the convention $0\log 0 = 0$, which is justified by continuity since $\lim_{t\to 0^+} -t\log t = 0$.

- To show that H is concave, let $P_1, P_2 \in \mathcal{P}(\mathcal{X})$, and let $\lambda \in [0,1]$. Define the convex combination:

$$P = \lambda P_1 + (1-\lambda)P_2$$

- The goal is to show that:

$$H(P) \geq \lambda H(P_1) + (1-\lambda)H(P_2)$$

- Define the function $f(t) = -t\log t$ for $t \in [0,1]$. Note that:

$$f''(t) = -\frac{1}{t} < 0 \quad \text{for } t \in (0,1]$$

Hence, $f(t)$ is strictly concave on $(0,1]$ and continuous on $[0,1]$, so it is concave on $[0,1]$.

- For each $x \in \mathcal{X}$, define:

$$P(x) = \lambda P_1(x) + (1-\lambda)P_2(x)$$

Then by concavity of f:

$$f(P(x)) \geq \lambda f(P_1(x)) + (1-\lambda)f(P_2(x))$$

or equivalently,

$$-P(x)\log P(x) \geq -\lambda P_1(x)\log P_1(x) - (1-\lambda)P_2(x)\log P_2(x)$$

- Summing over all $x \in \mathcal{X}$, we obtain:

$$H(P) = \sum_{x\in\mathcal{X}} -P(x)\log P(x) \geq \lambda\sum_{x\in\mathcal{X}} -P_1(x)\log P_1(x) + (1-\lambda)\sum_{x\in\mathcal{X}} -P_2(x)\log P_2(x)$$

$$= \lambda H(P_1) + (1-\lambda)H(P_2)$$

- Therefore, the Shannon entropy function is concave over the probability simplex:

$$H(\lambda P_1 + (1-\lambda)P_2) \geq \lambda H(P_1) + (1-\lambda)H(P_2)$$

(20250102#10)

In layman terms, describe AEP along with two simple examples - one where AEP holds true for the entire set of strings and the other where this doesn't happen:

Asymptotic Equipartition Property (AEP)

Almost all probability is concentrated on a set; each of whose elements has roughly equal probability.

Two examples: $A = \{0, 1\}$

1. X_n i.i.d. Bernoulli random variable Ber(1/2). Take any string → probability $P = 1/2^n$. AEP holds true for the entire set. All strings have equal probability.
2. X_n Ber(p) i.i.d.

   Strings with roughly nP 1s capture almost all probability. Total number of such strings:

   $$\binom{n}{k} p^k (1 - np)^{n-k}$$

   where $k = np$ roughly. All elements where $k \approx np$ has equal probability of occurrence.

(20250102#11)

Define δ-typical sets:

Let $(X_1, X_2, \ldots, X_n) \sim P_X^n$ be a sequence of i.i.d. random variables drawn from a discrete memoryless source with distribution P_X. For $\delta > 0$, the δ-typical set $A(n, \delta)$ is defined as:

$$A(n, \delta) = \left\{ x^n \in \mathcal{X}^n : \left| -\frac{1}{n} \log P_{X^n}(x^n) - H(X) \right| \leq \delta \right\}$$

where:

$$P_X(x^n) = \prod_{i=1}^{n} P_X(x_i), \quad \text{and} \quad H(X) = -\sum_{x \in \mathcal{X}} P_X(x) \log P_X(x)$$

$(x_1, x_2, \ldots, x_n)$ is δ-typical $\implies$

Let $(X_1, X_2, \ldots, X_n) \sim P_X^n$, where the source is memoryless (i.i.d.). The joint distribution is:

$$P_{X_1, \ldots, X_n}(x_1, \ldots, x_n) = \prod_{i=1}^{n} P_X(x_i)$$

Taking the logarithm of the joint distribution:

$$\log P_{X_1,\ldots,X_n}(x_1,\ldots,x_n) = \log\left(\prod_{i=1}^{n} P_X(x_i)\right)$$

$$= \sum_{i=1}^{n} \log P_X(x_i)$$

Using this, we have

$$A(n,\delta) = \left\{ x^n \in \mathcal{X}^n : \left| -\frac{1}{n}\sum_{i=1}^{n} \log P_X(x_i) - H(P_X) \right| \leq \delta \right\}$$

(20250102#12)

Show some of the properties of δ-typical sets?

Let $X_1, X_2, \ldots, X_n \sim P_X$ be i.i.d. random variables. The δ-typical set is defined as:

$$A(n,\delta) = \left\{ (x_1,\ldots,x_n) \in \mathcal{X}^n : \left| -\frac{1}{n} \log P_{X^n}(x_1,\ldots,x_n) - H(X) \right| < \delta \right\}$$

We prove the following key properties:

1. Probability Bounds for Typical Sequences

For any $(x_1,\ldots,x_n) \in A(n,\delta)$, since the source is memoryless:

$$P_X^n(x_1,\ldots,x_n) = \prod_{i=1}^{n} P_X(x_i)$$

Taking logs and using the definition of $A(n,\delta)$, we get:

$$\left| -\frac{1}{n} \log P_{X^n}(x_1,\ldots,x_n) - H(X) \right| < \delta$$

$$\Rightarrow H(X) - \delta < -\frac{1}{n} \log P_{X^n}(x_1,\ldots,x_n) < H(X) + \delta$$

Multiplying by $-n$ and exponentiating:

$$2^{-n(H(X)+\delta)} < P_X^n(x_1,\ldots,x_n) < 2^{-n(H(X)-\delta)}$$

2. Upper Bound on Size of Typical Set

The total probability of the typical set is at most 1, so:

$$1 \geq \sum_{x^n \in A(n,\delta)} P_X^n(x^n) > |A(n,\delta)| \cdot 2^{-n(H(X)+\delta)}$$

$$\Rightarrow |A(n,\delta)| < 2^{n(H(X)+\delta)}$$

3. Typical Set Has High Probability

Let $Z_i = \log \frac{1}{P_X(X_i)}$, so that $\mathbb{E}[Z_i] = H(X)$. Then:

$$-\frac{1}{n} \log P_{X^n}(X_1, \dots, X_n) = \frac{1}{n} \sum_{i=1}^{n} Z_i$$

By the Weak Law of Large Numbers:

$$\frac{1}{n} \sum_{i=1}^{n} Z_i \xrightarrow{P} H(X) \Rightarrow P\left(\left| -\frac{1}{n} \log P_{X^n}(X_1, \dots, X_n) - H(X) \right| > \delta \right) \to 0$$

$$\Rightarrow P\left((X_1, \dots, X_n) \in A(n,\delta)\right) \to 1$$

Almost all probability is concentrated on this δ-typical set with probability roughly $2^{-nH(P_X)}$.

(20250102#13)

Prove Shannon's source coding theorem for lossless compression:

Typically in IT proofs we have two parts - achievability/direct part and converse (can't do better than the achieved) part. We'll have to show them both to prove the theorem.

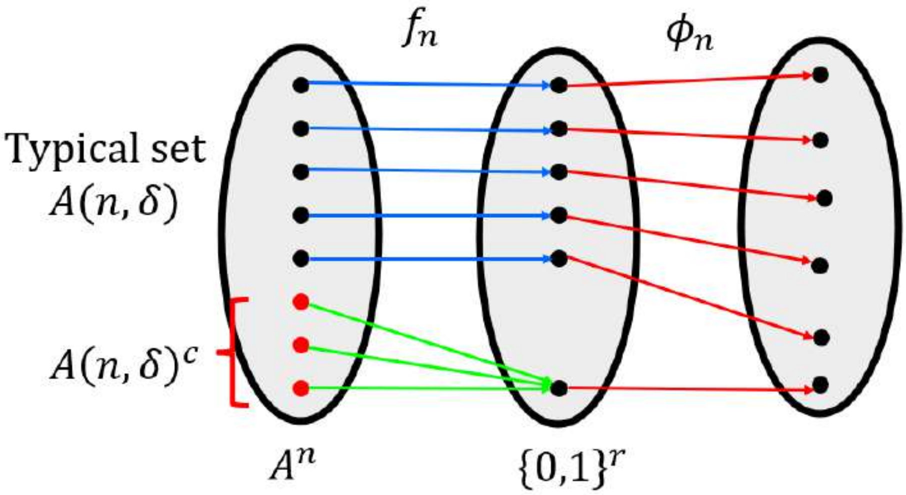
Let $\{X_i\}_{i=1}^{\infty}$ be a discrete memoryless source (DMS) with distribution P_X and entropy $H(X)$. Then:

Theorem. For every $\epsilon > 0$, there exists an n-length block source encoder-decoder pair (f_n, ϕ_n) with:

$$\frac{r(n,\epsilon)}{n} \leq H(P_X) + \delta \quad \text{and} \quad \Pr(\phi_n(f_n(X^n)) \neq X^n) \leq \epsilon$$

for large enough n, where $r(n,\epsilon)$ is the number of bits used for compression.

Achievability Proof



Let $A(n, \delta)$ be the δ-typical set:

$$A(n,\delta) = \left\{ x^n \in \mathcal{X}^n : \left| -\frac{1}{n} \log P_{X^n}(x^n) - H(X) \right| < \delta \right\}$$

Step 1: Choose $r = \lceil \log |A(n,\delta)| \rceil \leq \log_2 A(n,\delta) + 1$. Define a code only for typical sequences $x^n \in A(n,\delta)$. Since

$$|A(n,\delta)| \leq 2^{n(H(X)+\delta)},$$

we can assign binary codewords of length at most $r = \lceil n(H(X)+\delta) \rceil$ to each.

Step 2: Error probability Declare an error for sequences outside $A(n,\delta)$. Then:

$$\Pr(\text{error}) = \Pr(X^n \notin A(n,\delta)) \leq \epsilon,$$

for sufficiently large n, due to the weak law of large numbers.

Step 3: Rate bound The rate is:

$$\frac{r}{n} \leq H(X) + \delta + \frac{1}{n} \to H(X) + \delta$$

as $n \to \infty$. Hence,

$$\limsup_{n\to\infty} \frac{r(n,\epsilon)}{n} \leq H(X)$$

Converse

Suppose we have a sequence of source codes with rate $R = \frac{r(n)}{n} < H(X) - \delta$. Then, by Kraft's inequality and standard converse arguments, we get that the typical set cannot be fully covered, leading to:

$$\Pr(\text{decoding error}) \geq \Pr(X^n \in A(n,\delta)) \to 1 \quad \text{as } n \to \infty$$

Thus, achieving arbitrarily small error requires:

$$\liminf_{n\to\infty} \frac{r(n)}{n} \geq H(X)$$

Conclusion

Combining achievability and converse:

$$\lim_{n\to\infty} \inf \left\{ \frac{r(n,\epsilon)}{n} : e(f_n, \phi_n) \leq \epsilon \right\} = H(X)$$

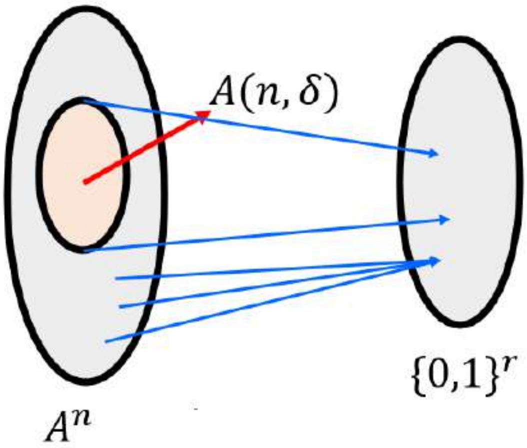
(20250107#14)

Prove this theorem:

$$\lim_{n\to\infty} \frac{r(n,\epsilon)}{n} \to_{n\to\infty} H(P_X), \quad \forall\epsilon \in (0,1)$$

Theorem: Let P_X be the probability distribution of a discrete memoryless source (DMS). Then for every $\epsilon \in (0,1)$, the minimum number of bits per symbol needed to encode source sequences of length n with error probability at most ϵ satisfies

$$\lim_{n\to\infty} \frac{r(n,\epsilon)}{n} = H(P_X)$$



Achievability (Direct Part)

Define the δ-typical set:

$$A(n,\delta) = \left\{ x^n \in \mathcal{X}^n : \left| -\frac{1}{n}\log P_{X^n}(x^n) - H(P_X) \right| \le \delta \right\}$$

Then:

- $|A(n,\delta)| \le 2^{n(H(P_X)+\delta)}$
- $P(A(n,\delta)) \to 1$ as $n \to \infty$
- For all $x^n \in A(n,\delta)$,

$$2^{-n(H(P_X)+\delta)} \le P_{X^n}(x^n) \le 2^{-n(H(P_X)-\delta)}$$

Construct code: Assign distinct binary codewords of length

$$r = \lceil \log |A(n,\delta)| \rceil$$

to each $x^n \in A(n,\delta)$. Ignore other sequences.

Error: Only sequences outside $A(n, \delta)$ cause decoding errors. Thus:

$$e(f_n, \phi_n) = P(A(n, \delta)^c) \leq \epsilon \quad \text{for large } n$$

Rate bound:

$$\frac{r(n, \epsilon)}{n} \leq \frac{r}{n} \leq \frac{\log |A(n, \delta)| + 1}{n} \leq H(P_X) + \delta + \frac{1}{n}$$

Taking lim sup,

$$\limsup_{n \to \infty} \frac{r(n, \epsilon)}{n} \leq H(P_X) + \delta$$

Since this holds for all $\delta > 0$,

$$\limsup_{n \to \infty} \frac{r(n, \epsilon)}{n} \leq H(P_X)$$

Converse

Suppose we have an (n, r) binary code with error probability $\leq \epsilon$. Let $\mathcal{C} \subseteq \mathcal{X}^n$ be the set of sequences decoded correctly.

Then:

$$|\mathcal{C}| \leq 2^r, \quad P_{X^n}(\mathcal{C}) \geq 1 - \epsilon$$

From AEP, for large n,

$$P(A(n, \delta)) \to 1 \Rightarrow P(\mathcal{C} \cap A(n, \delta)) \geq 1 - \epsilon - \eta$$

For all $x^n \in A(n, \delta)$,

$$P_{X^n}(x^n) \leq 2^{-n(H(P_X) - \delta)} \Rightarrow P(\mathcal{C} \cap A(n, \delta)) \leq |\mathcal{C} \cap A(n, \delta)| \cdot 2^{-n(H(P_X) - \delta)}$$

Hence:

$$|\mathcal{C} \cap A(n, \delta)| \geq (1 - \epsilon - \eta) \cdot 2^{n(H(P_X) - \delta)} \Rightarrow 2^r \geq (1 - \epsilon - \eta) \cdot 2^{n(H(P_X) - \delta)}$$

Taking log:

$$r \geq n(H(P_X) - \delta) + \log(1 - \epsilon - \eta) \Rightarrow \frac{r}{n} \geq H(P_X) - \delta + \frac{\log(1 - \epsilon - \eta)}{n}$$

Taking lim inf,

$$\liminf_{n \to \infty} \frac{r(n, \epsilon)}{n} \geq H(P_X) - \delta \Rightarrow \liminf_{n \to \infty} \frac{r(n, \epsilon)}{n} \geq H(P_X)$$

Conclusion

Combining both parts:

$$\lim_{n \to \infty} \frac{r(n, \epsilon)}{n} = H(P_X) \quad \forall \epsilon \in (0, 1)$$

(20250107#15)

Explain some ideas associated with block source coding:

1. **Compression with Error:** Allowing a small error probability $\epsilon > 0$, we can achieve significant compression. The number of bits $r(n, \epsilon)$ needed grows linearly with block length n. The compression ratio is characterized by the growth rate:

$$\lim_{n\to\infty} \frac{r(n,\epsilon)}{n} = H(P_X)$$

2. **Simplification in the Limit:** Though the encoding/decoding process is complex at finite n, the asymptotic analysis leads to a simple and universal result:

$$\text{Rate} \to H(P_X) \quad \text{as } n \to \infty$$

3. **Entropy as Growth Rate:** Entropy $H(P_X)$ captures the intrinsic randomness of the source. It governs the rate of growth of the number of typical sequences. Remarkably, this rate is independent of the target error probability ϵ, as long as $\epsilon \in (0, 1)$.

4. **Robustness to ϵ:** The coding rate $\frac{r(n,\epsilon)}{n}$ converges to $H(P_X)$ regardless of the exact value of $\epsilon \in (0, 1)$. The result is robust to the choice of error level, making the theorem practically useful.

5. **Asymptotic Equipartition Property (AEP):** AEP justifies using $\log \frac{1}{P_X(x)}$ as a natural measure of information. For a DMS,

$$-\frac{1}{n} \log P(X^n) \xrightarrow{a.s.} H(P_X)$$

so the source sequences concentrate in a typical set where all sequences have nearly equal probability.

6. **Law of Large Numbers:** The AEP is a consequence of the weak law of large numbers. It ensures that empirical averages (like information per symbol) converge to expected values.

7. **Every Equality is Two Inequalities:** The full source coding theorem is an equality in the limit. Its proof consists of two parts:
   - **Direct part:** Construct a code achieving rate $\leq H(P_X) + \delta$
   - **Converse part:** Show any code with error $\leq \epsilon$ must have rate $\geq H(P_X) - \delta$

8. **Buffering Room with $H(P_X) \pm \delta$:** Since exact equality can't be reached for finite n, we give ourselves some room using a small margin $\delta > 0$. This flexibility enables both the code construction and analysis.

9. **Likelihood Ratios:** Many results (e.g., hypothesis testing, typicality) involve comparing likelihoods. The ratio
$$\frac{P(x^n)}{Q(x^n)}$$
plays a central role in relative entropy and statistical inference.

10. **Relative Entropy:** The Kullback-Leibler divergence
$$D(P\|Q) = \sum_x P(x) \log \frac{P(x)}{Q(x)}$$
measures the inefficiency of assuming distribution Q when the true distribution is P. It appears in many converse bounds and reflects how far a distribution is from being typical under Q.

(20250107#16)

Motivate the definition of Kullback Leibler divergence using the notion of Q-weights:

We begin with the concept of a *minimum cardinality of sets* containing most of the probability mass:
$$|B_n| = \sum_{(x_1,\ldots,x_n)\in B_n} 1$$
where B_n is a subset of $\mathcal{X}^n$ such that
$$P(B_n) \geq 1 - \epsilon$$
This reflects the smallest number of sequences needed to cover most of the probability under P.

Generalization with Another Distribution Q

Now, we generalize this setup by introducing another probability mass function (PMF) Q over $\mathcal{X}$.

- For a sequence $(x_1, \ldots, x_n)$, define its Q-weight as:
$$Q(x_1, \ldots, x_n) = \prod_{i=1}^{n} Q(x_i)$$

- For a set $C_n \subseteq \mathcal{X}^n$, its total Q-weight is:
$$Q(C_n) = \sum_{(x_1,\ldots,x_n)\in C_n} Q(x_1, \ldots, x_n)$$

Minimum Q-Weighted Set Covering Most of P

Define

$$w(n, \epsilon) = \min\{Q(S) : P(S) \geq 1 - \epsilon\}$$

That is, $w(n, \epsilon)$ is the minimum Q-weight of a set S that captures at least $1 - \epsilon$ probability mass under P.

Theorem (Information Spectrum)

Let P and Q be distributions on $\mathcal{X}$ with $P \ll Q$. Then,

$$\lim_{n\to\infty} \frac{1}{n} \log w(n, \epsilon) = -D(P\|Q), \qquad \forall \epsilon \in (0, 1)$$

This result links the asymptotic behavior of weighted typical sets to the relative entropy between P and Q. It tells us how ”inefficient” it is to cover a typical P-set using Q-mass.

(20250107#17)

Show the application of minimum Q-weighted set that maximizes the P probability of a set in the context of hypothesis testing:

Consider the binary hypothesis testing problem based on observing $X_1, X_2, \ldots, X_n \in \mathcal{X}$:

- Null hypothesis: $H_0 : X_1, \ldots, X_n \overset{\text{iid}}{\sim} P$
- Alternative hypothesis: $H_1 : X_1, \ldots, X_n \overset{\text{iid}}{\sim} Q$

We define a decision rule using a subset $B_n \subseteq \mathcal{X}^n$:

- If $(X_1, \ldots, X_n) \in B_n$, declare H_0 (i.e., output 0)
- If $(X_1, \ldots, X_n) \in B_n^c$, declare H_1 (i.e., output 1)

Types of Error

- **False alarm probability (Type I error):**

  $$\alpha_n = P(B_n^c) = \text{Probability of declaring } H_1 \text{ when } H_0 \text{ is true}$$

  Typically, we require $\alpha_n \leq \epsilon$, for some small $\epsilon > 0$.

- **Missed detection probability (Type II error):**

  $$\beta_n = Q(B_n) = \text{Probability of declaring } H_0 \text{ when } H_1 \text{ is true}$$

Asymptotic Error Exponent

Define the minimum Type II error under a constraint on Type I error:

$$w(n, \epsilon) = \min \{Q(B_n) : P(B_n^c) \leq \epsilon\}$$

Theorem (Stein's Lemma): For any fixed $0 < \epsilon < 1$,

$$\lim_{n\to\infty} \frac{1}{n} \log w(n,\epsilon) = -D(P\|Q)$$

Interpretation

- The missed detection probability $w(n,\epsilon)$ decays exponentially at a rate determined by the Kullback–Leibler divergence $D(P\|Q)$.
- **Direct Part:** There exists a sequence of tests such that the missed detection probability satisfies
$$w(n,\epsilon) \lessapprox 2^{-nD(P\|Q)}$$
- **Converse Part:** No test can achieve a faster exponential decay; that is,
$$w(n,\epsilon) \gtrapprox 2^{-nD(P\|Q)}$$
- Thus, the exponential decay rate of the Type II error is precisely determined by the relative entropy between P and Q.

(20250107#18)

Give an example for variable length scheme and find its expected length:

Coding Scheme

We use the concept of δ-typical sets denoted by $A(n,\delta)$ for a discrete memoryless source with distribution P_X over a finite alphabet $\mathcal{A}$. The coding scheme works as follows:

- For any source sequence $x^n = (x_1, \ldots, x_n)$, check if $x^n \in A(n,\delta)$.
- If $x^n \in A(n,\delta)$, prepend a leading bit '1' and encode using a fixed-length code of length approximately $nH(P_X) + n\delta$.
- If $x^n \notin A(n,\delta)$, prepend a leading bit '0' and encode using a fixed-length code of length approximately $n(\log|\mathcal{A}| + 1)$.

Encoding Table

Case	**Probability**	**Codeword Length**
$x^n \in A(n,\delta)$ (Typical)	$\approx 1-\epsilon$	$1 + nH(P_X) + n\delta$
$x^n \notin A(n,\delta)$ (Atypical)	$\leq \epsilon$	$1 + n(\log|\mathcal{A}| + 1)$

Expected Length

Let $L(x^n)$ denote the codeword length for a sequence x^n. The expected length of the variable-length code is:

$$\mathbb{E}[L(X^n)] = P(X^n \in A(n,\delta)) \cdot (1 + nH(P_X) + n\delta) + P(X^n \notin A(n,\delta)) \cdot (1 + n(\log |\mathcal{A}| + 1))$$

Using the fact that $P(X^n \in A(n,\delta)) \geq 1 - \epsilon$ and $P(X^n \notin A(n,\delta)) \leq \epsilon$, we get:

$$\mathbb{E}[L(X^n)] \leq (1-\epsilon)(1 + nH(P_X) + n\delta) + \epsilon(1 + n(\log |\mathcal{A}| + 1))$$

Asymptotic Behavior

As $n \to \infty$, the impact of atypical sequences becomes negligible, and the expected length per symbol approaches the entropy rate:

$$\limsup_{n\to\infty} \frac{1}{n}\mathbb{E}[L(X^n)] \leq H(P_X) + \delta$$

(20250107#19)

What are some properties of entropy and relative entropy?

Let P and Q be probability mass functions defined on a finite alphabet $\mathcal{A}$. We consider their restriction to the δ-typical set, denoted by $A(n,\delta) \subseteq \mathcal{A}^n$.

1. Definition of Relative Entropy (KL Divergence)

$$D(P\|Q) = \sum_{a \in A(n,\delta)} P(a) \log \frac{P(a)}{Q(a)}$$

has geometrical interpretation of squared distance.

2. Properties of KL Divergence

- $D(P\|Q) \geq 0$ (non-negativity)
- Equality holds if and only if $P = Q$ almost everywhere (i.e., $P(a) = Q(a)$ for all $a \in A(n,\delta)$ with $P(a) > 0$)

3. Entropy of a Distribution

For a random variable $X \sim P$ taking values in $\mathcal{A}$, the entropy is defined as:

$$H(P) = H(X) = -\sum_{a \in \mathcal{A}} P(a) \log P(a)$$

4. Special Case: Uniform Distribution

Let P_U denote the uniform distribution over a finite set $\mathcal{A}$, i.e.,

$$P_U(a) = \frac{1}{|\mathcal{A}|}, \quad \forall a \in \mathcal{A}$$

Then the entropy of the uniform distribution is:

$$H(P_U) = \log |\mathcal{A}|$$

5. Entropy Bound

For any probability distribution P on $\mathcal{A}$,

$$H(P) \leq \log |\mathcal{A}|$$

with equality if and only if P is uniform on $\mathcal{A}$.

(20250107#20)

Prove the non-negativity of divergence using a tangent inequality:

Let P and Q be two probability distributions on a finite alphabet $\mathcal{A}$. The Kullback-Leibler (KL) divergence between P and Q is defined as:

$$D(P\|Q) = \sum_{a\in\mathcal{A}} P(a) \log \frac{P(a)}{Q(a)}$$

We aim to prove that:

$$D(P\|Q) \geq 0$$

with equality if and only if $P = Q$ pointwise.

Step 1: Use of the Inequality $\ln t \leq t - 1$

For all $t > 0$, the following inequality holds:

$$\ln t \leq t - 1$$

with equality if and only if $t = 1$.

This inequality is derived from the convexity of $\ln t$, whose tangent at $t = 1$ is:

$$\ln t \leq \ln 1 + (t-1) \cdot \frac{d}{dt} \ln t \Big|_{t=1} = 0 + (t-1)(1) = t - 1$$

Step 2: Apply the Inequality to the KL Expression

We change base to natural log for convenience (note that this only changes the units from bits to nats):

$$D(P\|Q) = \sum_{a \in \mathcal{A}} P(a) \ln \frac{P(a)}{Q(a)}$$

Let $t(a) = \frac{Q(a)}{P(a)}$. Then,

$$\ln \frac{P(a)}{Q(a)} = -\ln t(a) \geq -(t(a) - 1) = 1 - \frac{Q(a)}{P(a)}$$

Multiply both sides by $P(a)$ and sum:

$$D(P\|Q) = \sum_{a \in \mathcal{A}} P(a) \ln \frac{P(a)}{Q(a)} \geq \sum_{a \in \mathcal{A}} P(a) \left(1 - \frac{Q(a)}{P(a)}\right) = \sum_{a \in \mathcal{A}: P(a) > 0} (P(a) - Q(a))$$

Since $\sum_{a \in \mathcal{A}: P(a) > 0} P(a) = 1$, the sum becomes:

$$\sum_{a \in \mathcal{A}: P(a) > 0} (P(a) - Q(a)) = 1 - Q(\text{supp}(P)) - 1 \geq 0$$

where $\text{supp}(P)$ denotes the support of P, $\text{supp}(P) = \{a \in A : P(a) > 0\}$. Thus,

$$D(P\|Q) \geq 0$$

Step 3: Equality Condition

Equality holds if and only if $\ln \frac{P(a)}{Q(a)} = 0$ for all a with $P(a) > 0$, i.e.,

$$\frac{P(a)}{Q(a)} = 1 \quad \Rightarrow \quad P(a) = Q(a)$$

Hence, $D(P\|Q) = 0$ if and only if $P = Q$.

(20250109#21)

Prove $H(P_X) \leq \log |A|$ in two ways:

---

1. Using Block Coding and Source Coding Theorem

Let $X_1, \ldots, X_n$ be i.i.d. random variables with distribution P_X over a finite alphabet $\mathcal{A}$, where $|\mathcal{A}| = k$. Then the total number of possible strings of length n is $|\mathcal{A}^n| = |\mathcal{A}|^n$.

Any binary encoding of these strings must use at least

$$\lceil \log_2 |\mathcal{A}^n| \rceil = \lceil n \log_2 |\mathcal{A}| \rceil$$

bits. Hence, the average number of bits per symbol satisfies:

$$\frac{1}{n} \lceil \log_2 |\mathcal{A}^n| \rceil \leq \log_2 |\mathcal{A}| + \frac{1}{n}.$$

By Shannon's source coding theorem, the entropy satisfies:

$$H(P_X) \leq \lim_{n \to \infty} \frac{1}{n} \lceil \log_2 |\mathcal{A}^n| \rceil = \log_2 |\mathcal{A}|.$$

2. Using Non-negativity of KL Divergence

Let P be a distribution on $\mathcal{A}$, and let U be the uniform distribution on $\mathcal{A}$, i.e.,

$$U(a) = \frac{1}{|\mathcal{A}|}, \quad \forall a \in \mathcal{A}.$$

Consider the KL divergence between P and U:

$$D(P \| U) = \sum_{a \in \mathcal{A}} P(a) \log \frac{P(a)}{U(a)} \geq 0.$$

Since $U(a) = \frac{1}{|\mathcal{A}|}$, we can simplify:

$$D(P \| U) = \sum_{a \in \mathcal{A}} P(a) \log (P(a) \cdot |\mathcal{A}|) = \sum_{a \in \mathcal{A}} P(a) \log P(a) + \log |\mathcal{A}|.$$

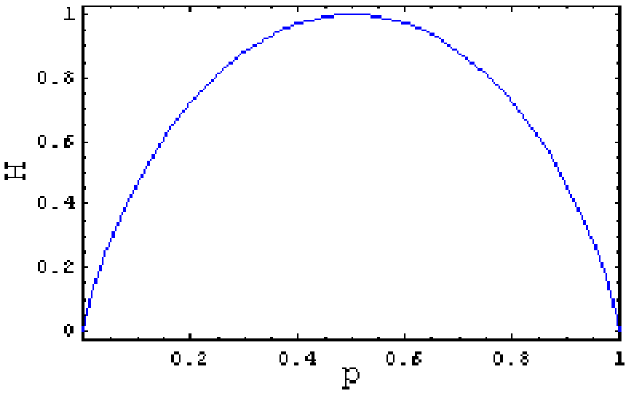
Hence,

$$-H(P_X) + \log |\mathcal{A}| \geq 0 \quad \Rightarrow \quad H(P_X) \leq \log |\mathcal{A}|.$$

Equality holds if and only if $P = U$, i.e., P is the uniform distribution on $\mathcal{A}$.

(20250109#22)

For binary alphabet with Bernoulli distribution Ber(p), find the binary entropy:



Consider a binary alphabet:

$$|\mathcal{A}| = 2, \quad \mathcal{A} = \{0, 1\}$$

Let $X \sim \text{Bern}(p)$, i.e., a Bernoulli random variable such that:

$$P(X = 1) = p, \quad P(X = 0) = 1 - p$$

The **binary entropy function**, denoted by $h(p)$, is defined as the entropy of the Bernoulli distribution:

$$h(p) := H(X) = -p \log_2 p - (1 - p) \log_2(1 - p)$$

for $0 < p < 1$, and we define $0 \log 0 := 0$ by convention.

Properties of the Binary Entropy Function

- $h(p)$ is symmetric about $p = \frac{1}{2}$: $h(p) = h(1 - p)$
- $h(p) \in [0, 1]$ for $p \in [0, 1]$
- $h(p)$ attains its maximum value of 1 at $p = \frac{1}{2}$
- $h(0) = h(1) = 0$

(20250109#23)

What will the entropy be in the scenario where the alphabet has ∞ cardinality?

Infinite Alphabet Case

Let the source alphabet have infinite cardinality: $|\mathcal{A}| = \infty$. In this setting:

- The entropy $H(P_X)$ can be either finite or infinite.
- This depends on the tail behavior of the probability distribution P_X over $\mathcal{A}$.

Exercise:

- Find a probability distribution P_X on $\mathcal{A} = \mathbb{N}$ (the natural numbers) such that the entropy $H(P_X)$ is infinite.
- **Hint:** Use a heavy-tailed distribution such as

$$P_X(k) = \frac{C}{k(\log k)^2}, \quad \text{for } k \geq 2,$$

  where C is a normalization constant to make $\sum_{k=2}^{\infty} P_X(k) = 1$.
- Show that this distribution is valid and that

$$H(P_X) = \sum_{k=2}^{\infty} -P_X(k) \log P_X(k) = \infty.$$

(20250109#24)

Prove that entropy doesn't change under one to one transformation of random variables:

Entropy Invariance Under One-to-One Transformation

Let X be a discrete random variable taking values in a finite or countable alphabet $\mathcal{A}$, with probability mass function $P_X(x)$. Let $f : \mathcal{A} \to \mathcal{B}$ be a one-to-one (injective) function, and define $Y = f(X)$. We want to show that:

$$H(Y) = H(X)$$

Proof:

- Since f is injective, the mapping from $x \in \mathcal{A}$ to $y = f(x) \in \mathcal{B}$ is one-to-one and invertible on its image.
- The probability mass function of Y is given by:

$$P_Y(y) = \mathbb{P}(Y = y) = \mathbb{P}(f(X) = y) = \mathbb{P}(X = f^{-1}(y)) = P_X(f^{-1}(y)).$$

- Since f is a bijection between $\mathcal{A}$ and $f(\mathcal{A}) \subseteq \mathcal{B}$, we have:

$$H(Y) = -\sum_{y \in f(\mathcal{A})} P_Y(y) \log P_Y(y) = -\sum_{x \in \mathcal{A}} P_X(x) \log P_X(x) = H(X).$$

$$\boxed{H(f(X)) = H(X) \quad \text{if } f \text{ is one-to-one}}$$

(20250109#25)

Explain briefly about joint entropy:

Joint Entropy

Let $(X, Y) \sim P_{XY}$ be a pair of discrete random variables defined on the product alphabet $\mathcal{A} \times \mathcal{B}$.

- The joint distribution P_{XY} can be viewed as a distribution P_Z on a new random variable $Z = (X, Y)$ taking values in the larger alphabet $\mathcal{A} \times \mathcal{B}$.
- The joint entropy of (X, Y) is defined as:

$$H(X, Y) := H(P_{XY}) = -\sum_{(x,y)\in\mathcal{A}\times\mathcal{B}} P_{XY}(x, y) \log P_{XY}(x, y)$$

- This quantity measures the total uncertainty (in bits) associated with the pair (X, Y) when the joint distribution P_{XY} is known.

(20250109#26)

Explain briefly about conditional entropy:

Conditional Entropy

Let $(X, Y) \sim P_{XY}$ be a pair of discrete random variables defined on the product alphabet $\mathcal{A} \times \mathcal{B}$.

- The **conditional entropy** of X given Y is defined as:

$$H(X|Y) := -\sum_{y\in\mathcal{B}} P_Y(y) \sum_{x\in\mathcal{A}} P_{X|Y}(x|y) \log P_{X|Y}(x|y)$$

This is the same as

$$H(X|Y) := \sum_{y\in\mathcal{B}} P_Y(y) H(X|Y = y)$$

and

$$H(X|Y) = -\sum_{(a,b)\in\mathcal{A}\times\mathcal{B}} P_{XY}(x, y) \log P_{X|Y}(x|y)$$

- Intuitively, this represents the expected number of bits needed to encode X when the value of Y is known.

- For each value $Y = b$, the encoder can choose a code optimized for the conditional distribution $P_{X|Y}(\cdot|b)$.

- This is analogous to using different block lengths or codebooks depending on the side information $Y = b$.

- In such scenarios, side information helps reduce uncertainty, often leading to $H(X|Y) \leq H(X)$.

(20250109#27)

State and prove chain rule of joint entropy:

Chain Rule of Entropy

- Let $(X, Y) \sim P_{XY}$ be a pair of discrete random variables over alphabets $\mathcal{A}$ and $\mathcal{B}$.

- The **joint entropy** of X and Y is defined as:

$$H(X,Y) := - \sum_{(x,y)\in\mathcal{A}\times\mathcal{B}} P_{XY}(x,y) \log P_{XY}(x,y)$$

- **Chain Rule of Entropy**:

$$H(X,Y) = H(X) + H(Y|X)$$

- This states that the total uncertainty (entropy) of the pair (X, Y) is equal to the uncertainty in X, plus the remaining uncertainty in Y after knowing X.

- **Proof**:

$$\begin{aligned}
H(X,Y) &= -\sum_{x,y} P_{XY}(x,y) \log P_{XY}(x,y) \\
&= -\sum_{x,y} P_{XY}(x,y) \log \left(P_X(x) P_{Y|X}(y|x)\right) \\
&= -\sum_{x,y} P_{XY}(x,y) \left[\log P_X(x) + \log P_{Y|X}(y|x)\right] \\
&= -\sum_{x,y} P_{XY}(x,y) \log P_X(x) - \sum_{x,y} P_{XY}(x,y) \log P_{Y|X}(y|x) \\
&= -\sum_{x} P_X(x) \log P_X(x) - \sum_{x} P_X(x) \sum_{y} P_{Y|X}(y|x) \log P_{Y|X}(y|x) \\
&= H(X) + H(Y|X)
\end{aligned}$$

General Chain Rule of Entropy

- Let $X_1, X_2, \ldots, X_n$ be discrete random variables.
- The **joint entropy** is defined as:

$$H(X_1, X_2, \ldots, X_n) := -\sum_{x_1,\ldots,x_n} P(x_1, \ldots, x_n) \log P(x_1, \ldots, x_n)$$

- **Chain Rule of Entropy**:

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

  where $H(X_1|X_1) := H(X_1)$.

- **Proof by Induction**:
  - **Base Case:** For $n = 2$,

$$H(X_1, X_2) = H(X_1) + H(X_2|X_1)$$

    which is the standard two-variable chain rule.
  - **Inductive Hypothesis:** Assume the result holds for $n = k$, i.e.,

$$H(X_1, \ldots, X_k) = \sum_{i=1}^{k} H(X_i | X_1, \ldots, X_{i-1})$$

  - **Inductive Step:** Consider $n = k + 1$. Then:

$$\begin{aligned} H(X_1, \ldots, X_k, X_{k+1}) &= H(X_1, \ldots, X_k) + H(X_{k+1}|X_1, \ldots, X_k) \\ &= \sum_{i=1}^{k} H(X_i|X_1, \ldots, X_{i-1}) + H(X_{k+1}|X_1, \ldots, X_k) \end{aligned}$$

  - Therefore, by induction, the chain rule holds for all $n \geq 2$:

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

(20250109#28)

Prove that

$$H(X_1, \ldots, X_n) \leq \sum_{i=1}^{n} H(X_i)$$

Interpretation: Just throw away all the side information in the RHS term of joint entropy expression.

Subadditivity of Entropy

- For discrete random variables $X_1, X_2, \ldots, X_n$, the following inequality holds:

$$H(X_1, X_2, \ldots, X_n) \leq \sum_{i=1}^{n} H(X_i)$$

- **Proof:**
  - By the chain rule of entropy, we have:

$$H(X_1, X_2, \ldots, X_n) = \sum_{i=1}^{n} H(X_i \mid X_1, \ldots, X_{i-1})$$

  - For each i, conditioning reduces entropy (or at least does not increase it):

$$H(X_i \mid X_1, \ldots, X_{i-1}) \leq H(X_i)$$

  - Therefore, summing over all i,

$$\sum_{i=1}^{n} H(X_i \mid X_1, \ldots, X_{i-1}) \leq \sum_{i=1}^{n} H(X_i)$$

  - Hence,

$$H(X_1, \ldots, X_n) \leq \sum_{i=1}^{n} H(X_i)$$

(20250109#29)

Prove the subadditivity of entropy using non-negativity of divergence:

Subadditivity of Entropy via KL Divergence

- Let $P_{X_1,\ldots,X_n}$ be the joint distribution of $(X_1, \ldots, X_n)$, and let $P_{X_1} \cdots P_{X_n}$ be the product of the marginals.

- The Kullback-Leibler (KL) divergence between the joint and product distributions is given by:

$$D(P_{X_1,\ldots,X_n} \,\|\, P_{X_1} \cdots P_{X_n}) = \sum_{x_1,\ldots,x_n} P(x_1, \ldots, x_n) \log \frac{P(x_1, \ldots, x_n)}{P(x_1) \cdots P(x_n)}$$

- Rearranging this:

$$D(P_{X_1,\ldots,X_n} \,\|\, P_{X_1} \cdots P_{X_n}) = -H(X_1, \ldots, X_n) + \sum_{i=1}^{n} H(X_i)$$

- Since KL divergence is always non-negative:

$$D(P_{X_1,\dots,X_n} \,\|\, P_{X_1} \cdots P_{X_n}) \geq 0$$

- Therefore,

$$-H(X_1, \dots, X_n) + \sum_{i=1}^{n} H(X_i) \geq 0 \Rightarrow H(X_1, \dots, X_n) \leq \sum_{i=1}^{n} H(X_i)$$

Thus for two variable case, we have this as well:

$$H(X|Y) \leq H(X)$$

Extra:

$$\mathbb{E}[X|Y] = g(y)$$

is in itself a random variable, which is a function of Y. But,

$$H(X|Y)$$

is just a number, not a random variable. Average conditional entropies given Y.

(20250109#30)

Motivate variable length instantaneous codes with an example:

Motivation for Instantaneous Codes

- Consider a source alphabet $\mathcal{A} = \{a, b, c\}$ and the following binary code assignments:

$$a \mapsto 0, \quad b \mapsto 01, \quad c \mapsto 011$$

- Suppose the receiver observes the string: 011
- Ambiguity arises:

$$\text{Is it } a \to 0,\ a \to 1,\ a? \quad \text{or } b \to 01,\ a \to 1? \quad \text{or } c \to 011?$$

- This shows that decoding cannot proceed symbol-by-symbol without looking ahead, making the decoding **non-instantaneous**.
- Now, consider a different code:

$$a \mapsto 0, \quad b \mapsto 10, \quad c \mapsto 11$$

- This is a **prefix-free code**: No codeword is a prefix of another.

- Upon observing a binary string (e.g., 0110), the decoder can parse unambiguously:

$$0 \to a, \quad 11 \to c, \quad 0 \to a \Rightarrow \text{decoded as } aca$$

- This is called an **instantaneous code** because decoding can be done without waiting for the rest of the string—symbol-by-symbol as soon as a valid codeword is detected.

Definition: A code is called **instantaneous** if it is prefix-free: No codeword is a prefix of any other codeword. This guarantees unique and immediate decodability.

(20250109#31)

What is a non-singular code?

A code C is a non-singular code if

$$x \neq y \implies C(x) \neq C(y)$$

(20250109#32)

What is a uniquely decodable code?

A code C is uniquely decodable if its extension C^* is non-singular. We're talking about extension via concatenation here.

(20250109#33)

When is a code prefix-free or instantaneous?

A code C is prefix-free or instantaneous if no codeword is a prefix of another codeword.

(20250109#34)

Explain Huffman's way of constructing prefix-free codes:

Huffman Coding: Constructing Optimal Prefix-Free Codes

- Huffman coding is a greedy algorithm that constructs an optimal prefix-free (instantaneous) code that minimizes expected codeword length for a given probability distribution.
- **Input:** A discrete random variable $X \in \mathcal{A}$ with probability mass function P_X.
- **Goal:** Assign binary codewords to each symbol in $\mathcal{A}$ such that the average length $\mathbb{E}[L]$ is minimized and the code is prefix-free.

Example:

Let the source alphabet be $\mathcal{A} = \{a, b, c, d, e\}$ with probabilities:

$$P_X = \{a : 0.4, \quad b : 0.2, \quad c : 0.2, \quad d : 0.1, \quad e : 0.1\}$$

1. Combine the two least probable symbols: $d(0.1)$ and $e(0.1)$ into a new node with probability 0.2.
2. Now we have: $a(0.4),\ b(0.2),\ c(0.2),\ de(0.2)$.
3. Combine any two least probable nodes (e.g., b, c or c, de), say $b(0.2)$ and $c(0.2)$ to get $bc(0.4)$.
4. Now: $a(0.4),\ de(0.2),\ bc(0.4)$
5. Combine $de(0.2)$ and $a(0.4)$ into $ade(0.6)$
6. Now: $bc(0.4),\ ade(0.6)$
7. Combine final two into root: $bcade(1.0)$

Assigning Codewords:

- Traverse the binary tree from root: Assign 0 to left branch, 1 to right branch.
- The leaf nodes (original symbols) now have prefix-free binary codewords:

$$a : 10, \quad b : 000, \quad c : 001, \quad d : 110, \quad e : 111$$

Expected Codeword Length:

$$\mathbb{E}[L] = 0.4 \cdot 2 + 0.2 \cdot 3 + 0.2 \cdot 3 + 0.1 \cdot 3 + 0.1 \cdot 3 = 2.6 \text{ bits}$$

Properties:

- Huffman coding produces an optimal prefix-free code.
- It achieves the minimal expected length among all prefix-free codes.
- Always satisfies: $H(P_X) \leq \mathbb{E}[L] < H(P_X) + 1$

(20250116#35)

Kraft's inequality gives a necessary and sufficient condition for existence of prefix-free (instantaneous) codes. Give the proof for Kraft inequality

Statement Let $\mathcal{C}$ be a binary prefix-free code with codeword lengths $\ell_1, \ell_2, \ldots, \ell_n$. Then, Kraft's inequality states:

$$\sum_{i=1}^{n} 2^{-\ell_i} \leq 1.$$

Conversely, if $\ell_1, \ldots, \ell_n$ are positive integers such that $\sum_{i=1}^{n} 2^{-\ell_i} \leq 1$, then there exists a binary prefix-free code with these codeword lengths.

Proof

Necessity (Prefix-free $\Rightarrow$ Inequality)

We model codewords as leaves of a binary tree, where each internal node has two children. Each codeword corresponds to a unique leaf, and no codeword is a prefix of another.

Let the codeword of length ℓ_i correspond to a leaf at depth ℓ_i.

For a binary tree with depth $L_{\max}$, maximum number of leaves possible is $2^{L_{\max}}$. So for such a binary tree, we can write

$$\sum_{i=1}^{L_{\max}} 2^{-L_{\max}} = 1$$

Each node at depth d has 2^d possible positions. Since codewords are prefix-free, the subtrees rooted at the codewords are disjoint. By having a codeword at level $\ell_i < L_{\max}$, we're removing $L_{\max} - \ell_i$ potential leaves of the full binary tree of depth $L_{\max}$. This is equivalent to the wastage of

$$2^{L_{\max} - \ell_i} \cdot 2^{L_{\max}} = 2^{-\ell_i}$$

of full tree's capacity.

Each codeword at depth ℓ_i "uses up" a fraction $2^{-\ell_i}$ of the tree's capacity. Therefore,

$$\sum_{i=1}^{n} 2^{-\ell_i} \leq 1.$$

Sufficiency (Inequality $\Rightarrow$ Prefix-free code exists)

Assume $\sum_{i=1}^{n} 2^{-\ell_i} \leq 1$ for integers $\ell_1, \ldots, \ell_n$. We construct a prefix-free binary code using a binary tree. Start with the full binary tree and assign codewords by taking the leftmost available leaves at depth ℓ_i for each i in order. Since the total sum is ≤ 1, there is enough room in the tree to assign disjoint leaves without violating the prefix condition.

Hence, a prefix-free binary code with these lengths exists.

□

(20250116#36)

Prove using Kraft's inequality that for any instantaneous code for a DMS satisfies

$$\mathbb{E}L \geq H(P_X) = H(X)$$

Setup Let X be a discrete memoryless source with finite alphabet $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ and probability distribution $P_X(x_i) = p_i$.

Let ℓ_i denote the length of the binary codeword assigned to symbol x_i. The expected codeword length is:

$$\mathbb{E}[L] = \sum_{i=1}^{n} p_i \ell_i.$$

Assume the code is **instantaneous** (prefix-free), so by Kraft's inequality:

$$\sum_{i=1}^{n} 2^{-\ell_i} \leq 1.$$

Objective We aim to show:

$$\mathbb{E}[L] \geq H(X) = -\sum_{i=1}^{n} p_i \log_2 p_i.$$

Proof Let us define:

$$q_i = \frac{2^{-\ell_i}}{K}, \quad \text{where } K = \sum_{j=1}^{n} 2^{-\ell_j} \leq 1.$$

Then $\{q_i\}$ defines a probability distribution because:

$$\sum_{i=1}^{n} q_i = \frac{1}{K} \sum_{i=1}^{n} 2^{-\ell_i} = 1.$$

Now compute the relative entropy $D(P\|Q)$ between distributions $P = \{p_i\}$ and $Q = \{q_i\}$:

$$D(P\|Q) = \sum_{i=1}^{n} p_i \log_2 \frac{p_i}{q_i}.$$

Substituting $q_i = 2^{-\ell_i}/K$:

$$D(P\|Q) = \sum_{i=1}^{n} p_i \log_2\left(\frac{p_i K}{2^{-\ell_i}}\right) = \sum_{i=1}^{n} p_i \log_2 p_i + \log_2 K + \sum_{i=1}^{n} p_i \ell_i.$$

Therefore:

$$D(P\|Q) = -H(X) + \log_2 K + \mathbb{E}[L].$$

Since $D(P\|Q) \geq 0$ and $\log_2 K \leq 0$, we get:

$$\mathbb{E}[L] \geq H(X) - \log_2 K \geq H(X).$$

Conclusion The expected length of any prefix-free (instantaneous) code satisfies:

$$\mathbb{E}[L] \geq H(X).$$

□

(20250116#37)

State the Shannon-Fano-Elias theorem:

Let X be a discrete memoryless source with finite alphabet $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$, and probability mass function $P_X(x_i) = p_i$, where $p_1 \geq p_2 \geq \cdots \geq p_n$.

Then there exists a prefix-free binary code assigning codewords of length

$$\ell_i = \left\lceil \log_2\left(\frac{1}{p_i}\right)\right\rceil \leq \log_2\left(\frac{1}{p_i}\right) + 1$$

to each symbol $x_i \in \mathcal{X}$, such that the expected codeword length satisfies:

$$H(X) \leq \mathbb{E}[L] \leq H(X) + 1,$$

where $H(X) = -\sum_{i=1}^{n} p_i \log_2 p_i$ is the Shannon entropy of the source.

Remarks

- We can build such a code if Kraft inequality holds.
- The code construction involves cumulative probabilities and ensures prefix-freeness.
- This result shows that it is possible to compress a source to nearly its entropy with an efficient prefix-free code.

- We can't do any better than entropy.

(20250116#38)

Show how Kraft inequality comes out of Shannon-Fano-Elias theorem:

$$\begin{aligned} L(a) &= \left\lceil \log \frac{1}{P_X(a)} \right\rceil \geq \log \frac{1}{P_X(a)} \\ 2^{L(a)} &\geq \frac{1}{P_X(a)} \\ P_X(a) &\geq 2^{-L(a)} \\ \sum_{a \in A} P_X(a) &\geq \sum_{a \in A} 2^{-L(a)} \\ &\implies \sum_{a \in A} 2^{-L(a)} \leq 1 \end{aligned}$$

we get the Kraft inequality.

(20250116#39)

How can $\left\lceil \log \frac{1}{P_X(a)} \right\rceil$ be interpreted?

In a prefix-free binary code, each symbol $a \in \mathcal{X}$ is assigned a binary codeword. This corresponds to a unique leaf in a binary tree. To decode a symbol, we traverse the tree from the root, asking a sequence of binary (yes-no) questions—one question per bit in the codeword.

Ideal vs Actual Code Length

The **ideal number of bits** needed to encode a symbol a with probability $P_X(a)$ is:

$$\log \frac{1}{P_X(a)},$$

which represents the amount of information contained in a.

However, since we can only use integer-length codewords in a binary code, the actual code length $\ell(a)$ must satisfy:

$$\ell(a) \geq \log \frac{1}{P_X(a)}.$$

To ensure this and maintain the prefix-free property, we choose:

$$\ell(a) = \left\lceil \log \frac{1}{P_X(a)} \right\rceil.$$

Relation to Kraft's Inequality

The Kraft inequality states that for any prefix-free binary code,

$$\sum_{a \in \mathcal{X}} 2^{-\ell(a)} \leq 1.$$

This ensures that the binary tree has enough "room" to accommodate all codewords without ambiguity.

Choosing $\ell(a) = \left\lceil \log \frac{1}{P_X(a)} \right\rceil$ guarantees that the code satisfies Kraft's inequality.

Expected Codeword Length and Entropy

The expected codeword length is:

$$\mathbb{E}[L] = \sum_{a \in \mathcal{X}} P_X(a)\ell(a) = \sum_{a \in \mathcal{X}} P_X(a) \left\lceil \log \frac{1}{P_X(a)} \right\rceil.$$

Since $\left\lceil \log \frac{1}{P_X(a)} \right\rceil \geq \log \frac{1}{P_X(a)}$, we have:

$$\mathbb{E}[L] \geq \sum_{a \in \mathcal{X}} P_X(a) \log \frac{1}{P_X(a)} = H(P_X).$$

Bottleneck Interpretation

The **bottleneck** in achieving $\mathbb{E}[L] = H(P_X)$ is the requirement that codeword lengths be integers. We can only ask whole binary questions when traversing the tree, so we must round up each $\log \frac{1}{P_X(a)}$ to the next integer. This rounding introduces a small inefficiency, making the expected length at least the entropy but not equal in general.

(20250116#40)

Motivate the need for source coding for dependent sources:

Consider an extreme example of a source that emits one of the following two sequences with equal probability:

- Sequence 1: $000\ldots0$ (length n)
- Sequence 2: $111\ldots1$ (length n)

Each sequence is chosen with probability 1/2. That is, the distribution over $(X_1, X_2, \ldots, X_n)$ is:

$$P(X_1 = \ldots = X_n = 0) = \frac{1}{2}, \quad P(X_1 = \ldots = X_n = 1) = \frac{1}{2}$$

Stationarity

This source is **stationary**, because the joint distribution of any shifted block of symbols is the same. That is, the distribution of $(X_1, \ldots, X_k)$ is identical to that of $(X_{t+1}, \ldots, X_{t+k})$ for all valid t, since all symbols in the sequence are always the same within a realization.

Redundancy and Entropy Rate

This source is **highly redundant**, because although each symbol can take values 0 or 1, once we know one symbol, we know the rest due to full dependence. Therefore, the joint entropy is:

$$H(X_1, \ldots, X_n) = 1 \text{ bit (only two possible sequences)}$$

So the entropy rate becomes:

$$\frac{1}{n} H(X_1, \ldots, X_n) = \frac{1}{n} \to 0 \quad \text{as } n \to \infty$$

This tells us that we only need 1 bit to represent any of the length-n sequences, even though each sequence is length n. Hence, the per-symbol cost is approaching zero.

Implication

If we are aware of the **dependence structure** among the symbols (i.e., that the symbols are perfectly correlated), we can achieve compression far better than encoding each symbol independently using $H(X)$, the marginal entropy.

Conclusion: For dependent sources, the entropy rate can be significantly lower than the entropy of a single symbol. Hence, understanding and leveraging statistical dependencies allows us to compress below $H(X)$, the marginal entropy.

(20250116#41)

Define δ-typical set for general dependent sources:

Let $X^n = (X_1, X_2, \dots, X_n) \in \mathcal{A}^n$ be a sequence of random variables with joint distribution $P_{X^n}(x^n)$. The δ**-typical set** $\mathcal{A}(n, \delta) \subset \mathcal{A}^n$ is defined as:

$$\mathcal{A}(n,\delta) = \left\{ x^n \in \mathcal{A}^n : \left| -\frac{1}{n} \log P_{X^n}(x^n) - \frac{1}{n} H(X^n) \right| \leq \delta \right\}$$

That is, $x^n \in \mathcal{A}(n, \delta)$ if the *empirical self-information* (log inverse probability) per symbol is close (within δ) to the entropy rate of the source.

Remarks

- For i.i.d. sources, this definition reduces to the standard definition of the weak typical set.
- For dependent sources, the joint distribution $P_{X^n}(x^n)$ captures the dependencies between symbols, so the typical set incorporates those dependencies.
- The size of $\mathcal{A}(n, \delta)$ is roughly $2^{H(X^n)}$, and the total probability mass $P_{X^n}(\mathcal{A}(n, \delta) \to 1$ as $n \to \infty$.

(20250116#42)

State the asymptotic equipartition property for dependent sources:

Let $(x_1, x_2, \dots, x_n) \in A^n$ be a realization of a stationary, ergodic source with joint distribution $P_{X^n}(x^n)$ and entropy rate $\mathcal{H} = \frac{1}{n} H(X^n)$. Define the δ-typical set:

$$A(n,\delta) = \left\{ x^n \in A^n : \left| -\frac{1}{n} \log P_{X^n}(x^n) - \mathcal{H} \right| \leq \delta \right\}$$

Then the **Asymptotic Equipartition Property (AEP)** states the following:

- **(Typicality)**: For every $\varepsilon > 0$, there exists N such that for all $n \geq N$,

$$\mathbb{P}\left(x^n \in A(n,\delta)\right) \geq 1 - \varepsilon$$

- **(Equipartition)**: For all $x^n \in A(n, \delta)$,

$$2^{-n(\mathcal{H}+\delta)} \leq P_{X^n}(x^n) \leq 2^{-n(\mathcal{H}-\delta)}$$

- **(Size of the typical set)**:

$$(1-\varepsilon) \cdot 2^{n(H-\delta)} \leq |A(n,\delta)| \leq 2^{n(H+\delta)}$$

Conclusion

The AEP implies that for large n, most of the probability mass is concentrated on a set of sequences of size approximately 2^{nH}, and all these sequences are almost equally likely, each with probability close to 2^{-nH}. This provides the foundation for source compression.

(20250116#43)

Give a counter example for a stationary sequence of random variables which won't obey AEP2:

Example 1: Binary Deterministic Source Let the source emit either of the following two infinite sequences with equal probability:

$$X_1, X_2, \dots = \begin{cases} 000\dots0 & \text{with probability } \frac{1}{2} \\ 111\dots1 & \text{with probability } \frac{1}{2} \end{cases}$$

Stationary: Yes. The process is stationary since the joint distribution of any block $(X_i, \dots, X_{i+k})$ is invariant under shifts.

Entropy rate: $H(X) = 0$, since after the first symbol, all others are deterministic.

AEP2: Satisfied. For any n,

$$\frac{1}{n} \log \frac{1}{P(X_1^n)} = \frac{1}{n} \log 2 \to 0 = H(X)$$

Example 2: One Deterministic + Uniform Part Now consider a source that emits one of 2^n+1 strings of length n:

- $000\dots0$: with probability $\frac{1}{2}$
- All other 2^n binary strings: each with probability $\frac{1}{2^{n+1}}$

Stationary: Yes, assuming the same distribution over sliding blocks.

AEP2: Not satisfied. Since,

$$\frac{1}{n} \log \frac{1}{P(X_1^n)} \in \left\{ \frac{1}{n}, 1 + \frac{1}{n} \right\}$$

With large probability, an element lies outside of the typical set. Typical set behavior is violated and hence AEP2 is not satisfied.

(20250116#44)

Find the entropy in this case: Consider the source that outputs binary strings of length n with the following distribution:

- The sequence $000\dots0$ (length n) has probability $P(x^{(1)}) = \frac{1}{2}$
- All other 2^n binary strings of length n (denoted $x^{(2)}, \dots, x^{(2^n+1)}$) are equally likely, each with probability $P(x^{(i)}) = \frac{1}{2^{n+1}}$ for $i = 2, \dots, 2^n + 1$

Entropy Computation

The entropy of the source over A^n is given by:

$$H(X^{(n)}) = -\sum_{i=1}^{2^n+1} P(x^{(i)}) \log_2 P(x^{(i)})$$

Breaking this sum into two parts:

- Contribution from $000\dots0$:
$$-\frac{1}{2}\log_2\frac{1}{2} = \frac{1}{2}$$
- Contribution from the remaining 2^n strings:
$$2^n \cdot \left(-\frac{1}{2^{n+1}}\log_2\frac{1}{2^{n+1}}\right) = 2^n \cdot \frac{1}{2^{n+1}} \cdot (n+1) = \frac{1}{2}(n+1)$$

Total entropy:

$$H(X^{(n)}) = \frac{1}{2} + \frac{1}{2}(n+1) = \frac{1}{2}(n+2)$$

Entropy Rate

$$\frac{H(X^{(n)})}{n} = \frac{1}{2}\left(1 + \frac{2}{n}\right)$$

Taking the limit as $n \to \infty$:

$$\lim_{n\to\infty} \frac{H(X^{(n)})}{n} = \frac{1}{2}$$

Interpretation

This source has entropy rate $\frac{1}{2}$, but it does *not* satisfy the Asymptotic Equipartition Property (AEP). The probability distribution is highly skewed:

- One sequence $(000\dots0)$ dominates the distribution with probability $\frac{1}{2}$.
- The remaining sequences each have exponentially small probability.

Hence, although the entropy rate is nonzero, the set of typical sequences (with approximately equal probability) does not exist in the usual sense required by the AEP.

(20250116#45)

How is it that in the example for previous question which can be equivalently viewed as the initial condition of choosing a pocket to decide the coin to toss have a long lasting consequence on the stochastic process?

Persistence of Initial Conditions in a Non-Ergodic Process

Consider the example where:

- With probability $\frac{1}{2}$, the output is the deterministic sequence $000\ldots0$ of length n.
- With probability $\frac{1}{2}$, the output is drawn uniformly from the remaining $2^n - 1$ binary sequences of length n (i.e., excluding $000\ldots0$).

This can be equivalently modeled as:

Pick one of two "pockets" at time $t = 0$:

- Pocket 1: always return the sequence $000\ldots0$.
- Pocket 2: choose a string uniformly at random from the rest of $\{0,1\}^n \setminus \{000\ldots0\}$.

Define a latent random variable $Z \in \{1,2\}$ that indicates the chosen pocket, with:

$$P(Z = 1) = \frac{1}{2}, \quad P(Z = 2) = \frac{1}{2}$$

Then the probability of a sequence $X^n = (X_1, \ldots, X_n)$ is given by:

$$P(X^n) = \sum_z P(X^n \mid Z = z) P(Z = z)$$

Why Initial Conditions Matter

- **Conditional Structure:** Given Z, the process is simple:
  - If $Z = 1$: all $X_i = 0$ deterministically.
  - If $Z = 2$: the sequence is uniformly distributed over $\{0,1\}^n \setminus \{000\ldots0\}$.

- **Unconditional Dependence:** Without knowing Z, the distribution over X^n mixes two vastly different distributions. The influence of Z persists for all n, violating the idea that distant parts of the sequence become independent.
- **Non-Ergodicity:** The process is not ergodic—the long-term behavior depends on the initial choice of Z. This lack of mixing implies that the Asymptotic Equipartition Property (AEP) fails.
- **Violation of AEP:** Since the sequences do not converge to a typical set with high probability, the property that most sequences have probability close to $2^{-nH(X)}$ does not hold.

Conclusion

This is an example of a **non-ergodic process** where a latent variable Z, chosen at the beginning, completely determines the global behavior of the sequence. This shows that:

> *Global statistical regularities (such as AEP) can break down when initial randomness has a persistent effect on the process.*

(20250116#46)

Give examples of some situations where we forget the initial conditions and the AEP will hold:

- Ergodic processes
- i.i.d processes
- Markov chain etc.

(20250116#47)

State the theorem for AEP for stationary and ergodic sources (McMillan, 1953):

Theorem (AEP for Stationary and Ergodic Sources (McMillan, 1953))**.** *Let $\{X_n\}_{n=1}^{\infty}$ be a stationary and ergodic stochastic process over a finite alphabet $\mathcal{X}$, with joint distribution $P_{X_1,X_2,\ldots}$. Define the entropy rate as:*

$$H = \lim_{n\to\infty} \frac{1}{n} H(X_1, X_2, \ldots, X_n)$$

Then, for P-almost every realization $(x_1, x_2, \ldots)$, the following limit holds:

$$\lim_{n\to\infty} -\frac{1}{n} \log P(X_1 = x_1, \ldots, X_n = x_n) = H$$

Equivalently,

$$-\frac{1}{n} \log P(X_1^n) \xrightarrow{a.s.} H$$

This means that for large n, the sequences x_1^n generated by the source are typical *in the sense that each has probability approximately 2^{-nH}, and the number of such sequences is approximately 2^{nH}.*

(20250116#48)

For a Markovian process, obtain the expression for empirical entropy:

Let $\{X_i\}_{i=1}^n$ be a first-order Markov process over a finite alphabet $\mathcal{X}$, with initial distribution $P_{X_1}(x_1)$ and transition probabilities $P_{X_i|X_{i-1}}(x_i|x_{i-1})$. Then the joint distribution of $(X_1, X_2, \ldots, X_n)$ is given by:

$$P_{X_1,\ldots,X_n}(x_1, \ldots, x_n) = P_{X_1}(x_1) \prod_{i=2}^{n} P_{X_i|X_{i-1}}(x_i|x_{i-1})$$

Taking the negative logarithm:

$$-\log P_{X_1,\ldots,X_n}(x_1, \ldots, x_n) = -\log P_{X_1}(x_1) - \sum_{i=2}^{n} \log P_{X_i|X_{i-1}}(x_i|x_{i-1})$$

Now divide both sides by n to normalize per symbol:

$$-\frac{1}{n} \log P_{X_1,\ldots,X_n}(x_1, \ldots, x_n) = \frac{1}{n} \left[-\log P_{X_1}(x_1)\right] + \frac{1}{n} \sum_{i=2}^{n} \left[-\log P_{X_i|X_{i-1}}(x_i|x_{i-1})\right]$$

Define:

$$f_1(X_1) = -\log P_{X_1}(X_1), \quad f_{12}(X_i, X_{i-1}) = -\log P_{X_i|X_{i-1}}(X_i|X_{i-1})$$

Then the expression becomes:

$$-\frac{1}{n} \log P_{X_1,\ldots,X_n}(x_1, \ldots, x_n) = \frac{1}{n} f_1(X_1) + \frac{1}{n} \sum_{i=2}^{n} f_{12}(X_i, X_{i-1})$$

This decomposition is useful in analyzing the entropy rate and applying asymptotic results like the AEP for Markov sources.

(20250116#49)

Explain Ergodic theorem via shift operators:

Let $w = (X_1, X_2, X_3, \ldots)$ be a realization of a stationary stochastic process $\{X_i\}_{i \geq 1}$, i.e., an infinite sequence. Define the **shift operator** T acting on the sequence w as follows:

$$T(w) = T(X_1, X_2, X_3, \ldots) = (X_2, X_3, X_4, \ldots)$$

Then, iterating the operator T, we have:

$$T^i(w) = (X_{i+1}, X_{i+2}, \ldots)$$

Now, consider a function f defined on the space of sequences, or equivalently on random variables of the form $f(X_i)$, or more generally $f(X_i, X_{i+1}, \ldots)$.

The **Ergodic Theorem** states: *If the process $\{X_i\}$ is stationary and ergodic, and f is integrable (i.e., $\mathbb{E}[|f(X_1)|] < \infty$), then:*

$$\frac{1}{n} \sum_{i=0}^{n-1} f(T^i w) \xrightarrow{\text{a.s.}} \mathbb{E}[f(X_1)]$$

This means that the time average (the average over shifts of the sequence w) converges almost surely to the ensemble average (expectation).

(20250116#50)

What is the challenge in showing AEP2?

Key Idea: In memoryless sources (i.i.d.), the probability of a sequence can be written as:

$$P(X_1, X_2, \ldots, X_n) = \prod_{i=1}^{n} P(X_i)$$

In this case, the analysis of AEP is straightforward.

However, for dependent sources such as Markov or more generally n-gram models, the probability of observing a symbol depends on a growing history.

1-gram model: (No history, i.i.d.)

$$P(X_i | X_{i-1}, X_{i-2}, \ldots) = P(X_i)$$

n-gram model: (Finite history of length $n-1$)

$$P(X_i | X_{i-1}, X_{i-2}, \ldots) = P(X_i | X_{i-1}, \ldots, X_{i-n+1})$$

General dependent source: (Possibly infinite memory)

$$P(X_1, X_2, \ldots, X_n) = P(X_1) \cdot P(X_2|X_1) \cdot P(X_3|X_1, X_2) \cdots P(X_n|X_1, \ldots, X_{n-1})$$

This leads to the log-probability expression:

$$-\frac{1}{n} \log P(X_1, X_2, \ldots, X_n) = -\frac{1}{n} \sum_{i=1}^{n} \log P(X_i|X_1, \ldots, X_{i-1})$$

Challenge: In this expression, the *conditioning set grows with* i, which means that the dependency structure becomes increasingly complex.

Unlike the i.i.d. case, we cannot reduce the analysis to a fixed distribution on a single symbol. Therefore, proving AEP2 requires strong assumptions, such as stationarity and ergodicity, to ensure convergence of:

$$\frac{1}{n} \log \frac{1}{P(X_1, \ldots, X_n)} \to H$$

almost surely, where H is the entropy rate of the process.

(20250116#51)

Give some properties of joint entropy and conditional entropy:

All of these holds true for stationary random variables.

1. $$H(X_n|X_1, \ldots, X_{n-1}) \leq H(X_{n-1}|X_1, \ldots, X_{n-2})$$

   assuming stationarity, and using the property that conditioning reduces entropy. Means that if we have more context available (more info on history), we can reduce the uncertainty of the present random variable.

2. $$\frac{H(X_1, \ldots, X_n)}{n} \geq H(X_n|X_1, \ldots, X_{n-1})$$

   using chain rule and first property.

3. $$\frac{H(X_1, \ldots, X_n)}{n} \leq \frac{H(X_1, \ldots, X_{n-1})}{n-1}$$

   which means given more information, we can compress better.

4. $$\lim_{n\to\infty} \frac{H(X_1, \ldots, X_n)}{n} = \lim_{n\to\infty} H(X_n|X_1, \ldots, X_{n-1})$$

(20250121#52)

Prove McMillan theorem for stationary and ergodic source:

Let $(X_1, X_2, \ldots, X_n) \in \mathcal{A}^n$. Define the δ-typical set $A(n, \delta) \subset \mathcal{A}^n$ as the set of sequences for which:

$$\left| -\frac{1}{n} \log P(X_1^n) - H \right| \leq \delta,$$

where H is the entropy rate of the source.

Theorem (McMillan)**.** *Let $\{X_i\}$ be a stationary and ergodic source. Then:*

$$\mathbb{P}[A(n,\delta)] \to 1 \quad \text{as } n \to \infty.$$

Proof. To prove this, we approximate the source by a finite memory process and pass to the limit.

Define a Markov approximation of order m:

$$Q_m(x_1^n) := P_{X_1^m}(x_1^m) \prod_{i=m+1}^{n} P_{X_i|X_{i-1},\ldots,X_{i-m}}(x_i|x_{i-1},\ldots,x_{i-m}).$$

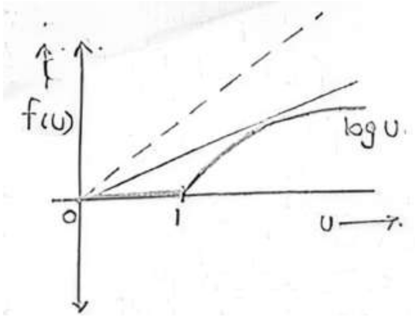
Now consider:

$$\frac{1}{n} \log \frac{1}{P(X_1^n)} = \underbrace{\left(\frac{1}{n} \log \frac{Q_m(X_1^n)}{P(X_1^n)} \right)}_{W_{m,n}} + \underbrace{\left(\frac{1}{n} \log \frac{1}{Q_m(X_1^n)} - H_m \right)}_{Z_{m,n}} + \underbrace{(H_m - H_n)}_{Y_{m,n}},$$

where $H_m := H(X_{m+1}|X_m, \ldots, X_1)$, and $H_n := -\frac{1}{n} \log P(X_1^n)$.

We will now analyze the convergence of each term.

Lemma. *If the source is stationary, then $Y_{m,n} \to 0$ as $m \to \infty, n \to \infty$.*



Proof. Since the source is stationary, the conditional entropy $H(X_{m+1}|X_m, \ldots, X_1)$ converges to the entropy rate. Therefore, the difference $H_m - H_n$ vanishes as $m, n \to \infty$. □

Lemma. $W_{m,n} \xrightarrow{L_1} 0$ *as* $n \geq m \to \infty$.

Proof. We aim to show:

$$\frac{1}{n}\mathbb{E}\left[\left|\log\frac{Q_m(X_1^n)}{P(X_1^n)}\right|\right] \to 0.$$

Using the inequality:

$$|\log u| \le 2au - \log u, \quad \text{for } u > 0,$$

with optimal $a = \frac{\log e}{e}$, we get:

$$\mathbb{E}\left[\left|\log\frac{Q_m}{P}\right|\right] \le \mathbb{E}\left[2a \cdot \frac{Q_m}{P} - \log\frac{Q_m}{P}\right].$$

Since $\sum_{x^n} P(x^n) \cdot \frac{Q_m(x^n)}{P(x^n)} = \sum_{x^n} Q_m(x^n) = 1$,

$$\mathbb{E}\left[\frac{Q_m}{P}\right] = 1,$$

and:

$$\mathbb{E}\left[\log\frac{Q_m}{P}\right] = H(P) - H(Q_m),$$

we obtain:

$$\frac{1}{n}\mathbb{E}[|\log Q_m - \log P|] \to 0.$$

Thus, $W_{m,n} \to 0$ in L_1. □

Lemma. *If the source is stationary and ergodic, then $Z_{m,n} \xrightarrow{a.s.} 0$ as $n \to \infty$ for fixed m.*

Proof. Let us expand:

$$\frac{1}{n}\log\frac{1}{Q_m(X_1^n)} = \frac{1}{n}\log\frac{1}{P(X_1^m)} + \frac{1}{n}\sum_{i=m+1}^{n}\log\frac{1}{P(X_i|X_{i-1},\ldots,X_{i-m})}.$$

Define:

$$Z_{m,n} = \frac{n-m}{n} \cdot \frac{1}{n-m}\sum_{i=m+1}^{n}\log\frac{1}{P(X_i|X_{i-1},\ldots,X_{i-m})} - H_m.$$

By the ergodic theorem, the time average converges almost surely to the expectation, hence:

$$Z_{m,n} \xrightarrow{a.s.} 0.$$

□

Putting everything together:

$$\frac{1}{n}\log\frac{1}{P(X_1^n)} \to H \quad \text{in probability.}$$

Hence,

$$\mathbb{P}[A(n,\delta)] \to 1 \quad \text{as } n \to \infty.$$

□

(20250121#53)

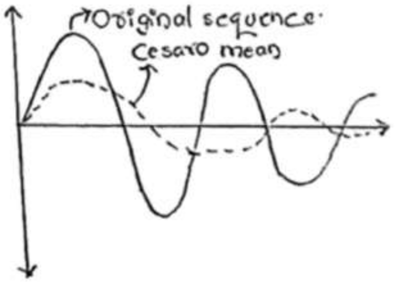
Explain Cesàro mean:

Let $\{a_n\}_{n\in\mathbb{N}}$ be a sequence of real (or complex) numbers. The **Cesàro mean** of the sequence $\{a_n\}$ is defined as the sequence $\{c_n\}_{n\in\mathbb{N}}$ given by

$$c_n = \frac{1}{n}\sum_{k=1}^{n} a_k. \tag{1}$$

This is the sequence of arithmetic means (or averages) of the first n terms of $\{a_k\}$.

Definition (Cesàro Summability):



A sequence $\{a_n\}$ is said to be **Cesàro summable** to L if its Cesàro mean c_n converges to L, i.e.,

$$\lim_{n\to\infty} c_n = \lim_{n\to\infty}\frac{1}{n}\sum_{k=1}^{n} a_k = L. \tag{2}$$

We write this as $a_n \xrightarrow{(C,1)} L$, where $(C,1)$ denotes Cesàro summability of order 1.

Key Property:

If $\lim_{n\to\infty} a_n = L$, then $\lim_{n\to\infty} c_n = L$ as well.

That is, ordinary convergence implies Cesàro convergence.

Note: The converse is not necessarily true. That is, Cesàro convergence does not imply ordinary convergence.

Example: Consider the sequence $a_n = (-1)^{n+1}$. This sequence does not converge in the usual sense. However, its Cesàro mean is:

$$c_n = \frac{1}{n}\sum_{k=1}^{n}(-1)^{k+1}. \tag{3}$$

This alternates between values close to 1/2 and hence:

$$\lim_{n\to\infty} c_n = \frac{1}{2}. \tag{4}$$

So a_n is Cesàro summable to 1/2.

(20250121#54)

State Breiman's theorem for a stationary and ergodic source:

Let $\{X_n\}_{n\in\mathbb{N}}$ be a stationary and ergodic stochastic process over a finite or countable alphabet $\mathcal{X}$, with joint distribution $P_{X_1,\ldots,X_n}$ and entropy rate

$$H = \lim_{n\to\infty} \frac{1}{n} H(X_1, X_2, \ldots, X_n).$$

Then, the normalized information content converges almost surely and in L_1 to the entropy rate:

$$\lim_{n\to\infty} -\frac{1}{n} \log P_{X_1,\ldots,X_n}(X_1, \ldots, X_n) = H \quad \text{a.s. and in } L_1. \tag{5}$$

Interpretation: For a stationary and ergodic source, the per-symbol code length under the true distribution converges to the entropy rate H both almost surely and in expected value.

Reference for Proof: A detailed proof of this result using martingale and ergodic arguments can be found in:

- R. E. Barron, "The strong ergodic theorem for densities: generalized Shannon-McMillan-Breiman theorem," *Annals of Probability*, vol. 13, no. 4, pp. 1292–1303, 1985.

(20250121#55)

Give entropy rate of a Markov source:

Consider a first-order Markov source $\{X_n\}_{n\geq 1}$ taking values in a finite alphabet $\mathcal{A}$, with transition probabilities

$$P_{ab} = P_{X_2|X_1}(b|a), \quad \text{for } a, b \in \mathcal{A}.$$

Assume the Markov chain is stationary with stationary distribution Π, satisfying:

$$\Pi = \Pi P, \quad \text{where } P = [P_{ab}]_{a,b\in\mathcal{A}}.$$

The entropy rate H of the source is given by the conditional entropy:

$$H = H(X_2|X_1) = \sum_{a\in\mathcal{A}} \Pi(a) \sum_{b\in\mathcal{A}} P_{X_2|X_1}(b|a) \log\left(\frac{1}{P_{X_2|X_1}(b|a)}\right).$$

Equivalently, this can be written as:

$$H = -\sum_{a\in\mathcal{A}} \Pi(a) \sum_{b\in\mathcal{A}} P_{X_2|X_1}(b|a) \log P_{X_2|X_1}(b|a).$$

Interpretation: The entropy rate of a stationary Markov source equals the expected uncertainty of the next symbol X_2 given the current symbol X_1, where the expectation is taken with respect to the stationary distribution Π.

(20250121#56)

How do we go about source compression when we don't know the true distribution P_X?

The key idea to be conveyed here is **universality**:

What if the true distribution P_X is unknown?

Let us return to the i.i.d. source model.

Suppose we observe a specific realization $x_1, x_2, \ldots, x_n$ of a source over a finite alphabet $\mathcal{A}$. Our goal is to compress it.

Naive approach:

Estimate the empirical distribution from the sequence and compress accordingly.

- Define the empirical distribution (type) as:

$$\tau(a; x^n) := \frac{1}{n}\sum_{i=1}^{n} \mathbf{1}\{x_i = a\}, \quad \forall a \in \mathcal{A}.$$

- $\tau(\cdot; x^n)$ is the empirical distribution of x^n.
- First, encode the number of occurrences of each symbol.

  Since $\sum_{a\in\mathcal{A}} n\tau(a; x^n) = n$, if we know the counts for $|\mathcal{A}| - 1$ symbols, the last one is determined.

  Each count can take values from 0 to n, i.e., $n + 1$ possibilities. Hence, the number of bits required is:

$$(|\mathcal{A}| - 1) \cdot \lceil \log(n + 1)\rceil.$$

- The complexity arises in encoding the exact sequence x^n among all sequences with the same empirical distribution.

The number of such sequences is given by the multinomial coefficient:

$$|\mathcal{T}_\tau| = \frac{n!}{\prod_{a\in\mathcal{A}}(n\tau(a;x^n))!}.$$

Apply Stirling's approximation:

$$k! \approx k^k e^{-k}\sqrt{2\pi k},$$

to estimate the total number of sequences in the type class:

$$\log\left(\frac{n!}{\prod_{a\in\mathcal{A}}(n\tau(a))!}\right) \approx \log(n!) - \sum_{a\in\mathcal{A}} \log((n\tau(a))!).$$

Using Stirling's approximation:

$$\log(n!) \approx n\log n - n + \frac{1}{2}\log(2\pi n), \quad \log((n\tau(a))!) \approx n\tau(a)\log(n\tau(a)) - n\tau(a) + \frac{1}{2}\log(2\pi n\tau(a)).$$

Substituting and simplifying:

$$\log|\mathcal{T}_\tau| \approx n\log n - n - \sum_{a\in\mathcal{A}} [n\tau(a)\log(n\tau(a)) - n\tau(a)] + O(\log n).$$

This simplifies to:

$$\log|\mathcal{T}_\tau| \approx -n\sum_{a\in\mathcal{A}} \tau(a)\log\tau(a) + O(\log n) = nH(\tau) + O(\log n).$$

Therefore:

$$\boxed{\log|\mathcal{T}_\tau| \leq nH(\tau) + n\cdot O\left(\frac{\log n}{n}\right).}$$

This shows that the complexity of compression lies in identifying the particular sequence x^n among all the sequences of the same type.

(20250123#57)

Come up with a universal coding strategy for compressing X^n without knowing the true distribution:

Let $x^n = (x_1, x_2, \ldots, x_n)$ be a string over a finite alphabet $\mathcal{A}$.

We describe a universal coding strategy for compressing x^n without knowledge of the underlying distribution P_X.

Steps of the Encoding Procedure:

1. Identify the empirical distribution (type) of the sequence:

$$\tau(a; x^n) := \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}\{x_i = a\}, \quad \forall a \in \mathcal{A}.$$

2. Encode the empirical distribution $\tau(\cdot; x^n)$. Since the counts must sum to n, we only need to encode the counts of $|\mathcal{A}| - 1$ symbols, each requiring $\lceil \log(n+1) \rceil$ bits. Total:

$$(|\mathcal{A}| - 1) \cdot \lceil \log(n+1) \rceil \text{ bits.}$$

3. Resolve the ambiguity of which sequence corresponds to the given type:

   The total number of sequences with empirical distribution τ is:

$$|\mathcal{T}_\tau| = \frac{n!}{\prod_{a \in \mathcal{A}} (n\tau(a))!}.$$

   This is analogous to specifying one of the $\binom{n}{k}$ binary strings of weight k; here, it generalizes to multiple symbol counts. One can enumerate all such sequences and specify the index of x^n among them.

   Hence, encoding x^n given its type requires approximately:

$$\log \left(\frac{n!}{\prod_{a \in \mathcal{A}} (n\tau(a))!} \right) \approx nH(\tau) + o(n) \text{ bits,}$$

   where $H(\tau)$ is the entropy of the empirical distribution.

Remarks:

- The encoder does not require knowledge of P_X — only the observed sequence x^n is needed to compute $\tau(\cdot; x^n)$.
- The decoder also does not require P_X — it reconstructs x^n from the encoded type and the index among type class sequences.
- Hence, this scheme is **universal** for all i.i.d. sources.

- Total code length:

$$L_n(x^n) \leq (|\mathcal{A}| - 1)\lceil \log(n+1) \rceil + nH(\tau(\cdot; x^n)) + o(n).$$

- Divide by n:

$$\frac{L_n(x^n)}{n} \leq o(1) + H(\tau(\cdot; x^n)).$$

- For i.i.d. sources, $\tau(\cdot; x^n) \to P_X$ as $n \to \infty$ (by the law of large numbers).
- Taking expectation:

$$\lim_{n\to\infty} \mathbb{E}\left[\frac{L_n(x^n)}{n}\right] \leq \lim_{n\to\infty} \mathbb{E}\left[H(\tau(\cdot; x^n))\right].$$

- Since entropy is concave, Jensen's inequality gives:

$$\mathbb{E}[H(\tau)] \leq H(\mathbb{E}[\tau]) = H(P_X).$$

- The last inequality uses the fact that conditioning reduces entropy.

(20250123#58)

Show how conditioning reduces entropy in this context:

Let Z and Y be discrete random variables defined on finite alphabets $\mathcal{A}$ and $\mathcal{B}$ respectively, with joint distribution $P_{Z,Y}$. We aim to show:

$$H(Z|Y) \leq H(Z),$$

with equality if and only if Z and Y are independent.

Definitions and Setup:

- The marginal distribution of Z is

$$P_Z(z) = \sum_{y\in\mathcal{B}} P_{Z|Y}(z|y)P_Y(y).$$

- The (unconditional) entropy of Z is

$$H(Z) = -\sum_{z\in\mathcal{A}} P_Z(z)\log P_Z(z).$$

- The conditional entropy is the expected entropy of Z given each value of Y:
$$H(Z|Y) = \sum_{y \in \mathcal{B}} P_Y(y) H(Z|Y = y),$$
where
$$H(Z|Y = y) = -\sum_{z \in \mathcal{A}} P_{Z|Y}(z|y) \log P_{Z|Y}(z|y).$$

Main Idea:

Let us denote $P_{Z|Y=y} =: Q_y$. Then, the marginal distribution is the expectation:
$$P_Z = \sum_{y \in \mathcal{B}} P_Y(y) Q_y = \mathbb{E}[Q_Y].$$

Entropy as a Concave Function:

Entropy is a concave function on the probability simplex. That is, for any random distribution Q_Y, we have:
$$H\left(\mathbb{E}[Q_Y]\right) \geq \mathbb{E}\left[H(Q_Y)\right].$$

This is a direct application of Jensen's inequality:
$$H(Z) = H\left(\sum_{y \in \mathcal{B}} P_Y(y) P_{Z|Y=y}\right) \geq \sum_{y \in \mathcal{B}} P_Y(y) H(P_{Z|Y=y}) = H(Z|Y).$$

Conclusion:

$$H(Z|Y) \leq H(Z),$$
with equality if and only if $P_{Z|Y=y} = P_Z$ for all $y \in \mathcal{B}$, i.e., Z and Y are independent.

(20250123#59)

Define type of a sequence and type class with an example:

Let $\mathcal{A}$ be a finite alphabet, and let $x^n = (x_1, x_2, \ldots, x_n) \in \mathcal{A}^n$ be a sequence.

- The **type** (or empirical distribution) of x^n, denoted $\tau(\cdot, x^n)$, is the empirical probability mass function defined as:
$$\tau(a; x^n) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}\{x_i = a\}, \quad \forall a \in \mathcal{A}.$$
That is, $\tau(a; x^n)$ is the relative frequency of symbol a in the sequence x^n.

Set of Types:

- Define the set of all **types of sequences of length** n as:

$$\mathcal{T}_n := \{\tau(\cdot, x^n) : x^n \in \mathcal{A}^n\} \subseteq \mathcal{P}(\mathcal{A}),$$

where $\mathcal{P}(\mathcal{A})$ is the set of all probability mass functions (PMFs) on $\mathcal{A}$.

Type Class:

- Given a type $\tau \in \mathcal{T}_n$, the **type class** associated with τ is defined as:

$$A_n(\tau) := \{x^n \in \mathcal{A}^n : \tau(\cdot, x^n) = \tau\}.$$

This is the set of all sequences of length n that have the same empirical distribution τ.

Example:

Let $\mathcal{A} = \{0, 1\}$ and consider $n \geq 1$. Define the type:

$$\tau(0) = \frac{1}{n}, \quad \tau(1) = 1 - \frac{1}{n}.$$

Then,

$$A_n(\tau) = \{x^n \in \{0,1\}^n : x^n \text{ contains exactly one 0 and } n-1 \text{ ones}\}.$$

That is, $A_n(\tau)$ consists of all binary strings of length n with exactly one zero.

(20250123#60)

Prove:

$$|\mathcal{T}_n| \leq (n+1)^{|\mathcal{A}|}$$

Let $\mathcal{A}$ be a finite alphabet with size $|\mathcal{A}| = k$. Then the number of distinct types (empirical distributions) of sequences of length n over $\mathcal{A}$ satisfies:

$$|\mathcal{T}_n| \leq (n+1)^{|\mathcal{A}|}$$

Proof:

- Let a type τ be an empirical distribution over $\mathcal{A}$, i.e., $\tau(a) = \frac{n_a}{n}$ for some nonnegative integers n_a satisfying:

$$\sum_{a \in \mathcal{A}} n_a = n, \quad \text{and} \quad n_a \in \{0, 1, \ldots, n\} \; \forall a \in \mathcal{A}.$$

- That is, a type corresponds to an integer vector $(n_a : a \in \mathcal{A})$ summing to n, which is a composition of n into $|\mathcal{A}|$ nonnegative integers.

- Each n_a can take $n+1$ possible values: $0, 1, \ldots, n$.

- But the constraint $\sum n_a = n$ restricts the number of valid combinations.

- Still, the total number of such integer vectors (types) is upper bounded by:

$$|\mathcal{T}_n| \leq (n+1)^{|\mathcal{A}|}.$$

This is because for each of the $|\mathcal{A}|$ symbols, the frequency can independently take at most $n+1$ values (from 0 to n), ignoring the constraint for an upper bound.

□

(20250123#61)

Explain about type class partition and for a binary case, find out the joint probability using empirical distributions:

- For any finite alphabet $\mathcal{A}$, the set of all sequences of length n, $\mathcal{A}^n$, can be partitioned based on their empirical distributions (types):

$$\bigcup_{\tau \in \mathcal{T}_n} A_n(\tau) = \mathcal{A}^n$$

where $A_n(\tau) = \{x^n \in \mathcal{A}^n : \tau(x^n) = \tau\}$, and $\mathcal{T}_n$ is the set of all possible types for sequences of length n. Thus,

$$\{A_n(\tau) : \tau \in \mathcal{T}_n\} \text{ forms a partition of } \mathcal{A}^n.$$

- **Binary Case:** Let $\mathcal{A} = \{0, 1\}$. Consider a sequence $x^n \in \{0,1\}^n$ that contains exactly k ones and $n-k$ zeros.

- Let the probability mass function be $P = (p(0), p(1))$. For an i.i.d. source, the probability of a specific sequence x^n with this composition is:

$$P_{X_1,\ldots,X_n}(x^n) = P^n(x^n) = p(0)^{n-k} p(1)^k$$

- Using logarithms:

$$\begin{aligned}
\log_2 P^n(x^n) &= (n-k)\log_2 p(0) + k \log_2 p(1) \\
&= n\left(\frac{n-k}{n}\log_2 p(0) + \frac{k}{n}\log_2 p(1)\right)
\end{aligned}$$

- Therefore, the probability of x^n can be expressed as:

$$P^n(x^n) = 2^{n\left(\frac{k}{n}\log_2 p(1) + \frac{n-k}{n}\log_2 p(0)\right)}.$$

- This expression highlights how the probability of a sequence depends only on its type τ, not on the specific order of symbols. All sequences in the same type class $A_n(\tau)$ have the same probability.

(20250123#62)

Let $x^n \in A^n$ be a sequence of type τ, and let P be the true i.i.d. source distribution. Then prove this lemma:

$$P^n(x^n) = 2^{-n[H(\tau)+D(\tau\|P)]}$$

Proof:

- Let $\mathcal{A}$ be the finite alphabet. The type τ of sequence x^n is the empirical distribution:

$$\tau(a; x^n) = \frac{N(a|x^n)}{n}, \quad \text{for each } a \in \mathcal{A}$$

  where $N(a|x^n)$ is the number of occurrences of symbol a in the sequence x^n.

- Since the source is i.i.d., the probability of $x^n = (x_1, \ldots, x_n)$ under distribution P is:

$$P^n(x^n) = \prod_{i=1}^{n} P(x_i) = \prod_{a\in\mathcal{A}} P(a)^{N(a|x^n)} = \prod_{a\in\mathcal{A}} P(a)^{n\tau(a)}$$

- Taking logarithms base 2:

$$\log_2 P^n(x^n) = \sum_{a\in\mathcal{A}} n\tau(a)\log_2 P(a) = n\sum_{a\in\mathcal{A}} \tau(a)\log_2 P(a)$$

- Use the following identity:

$$\sum_{a\in\mathcal{A}} \tau(a)\log_2 P(a) = \sum_{a\in\mathcal{A}} \tau(a)\log_2 \frac{\tau(a)}{P(a)} - \sum_{a\in\mathcal{A}} \tau(a)\log_2 \tau(a)$$

- Therefore:

$$\log_2 P^n(x^n) = -n\left(H(\tau) + D(\tau\|P)\right)$$

- Exponentiating both sides:

$$P^n(x^n) = 2^{-n[H(\tau)+D(\tau\|P)]}$$

- This completes the proof.

■

(20250123#63)

Corollary: If $P = \tau$, and $x^n \in A_n(\tau)$, then $\tau_n(x_1, \ldots, x_n) = 2^{-nH(\tau)}$. Show this:

Let $P = \tau$ and let $x^n \in A_n(\tau)$, i.e., the sequence $x^n = (x_1, \ldots, x_n)$ is of type τ. Then the probability of the sequence under distribution τ is:

$$\tau^n(x_1, \ldots, x_n) = 2^{-nH(\tau)}$$

Proof:

- Since x^n is of type τ, the number of occurrences of each symbol $a \in \mathcal{A}$ in x^n is $n\tau(a)$.
- The probability of x^n under the product distribution τ^n is:

$$\tau^n(x^n) = \prod_{a \in \mathcal{A}} \tau(a)^{n\tau(a)} = 2^{n \sum_{a \in \mathcal{A}} \tau(a) \log_2 \tau(a)} = 2^{-nH(\tau)}$$

- Therefore, the result follows. Note that it can also be shown using the result derived in the previous question and setting $D = 0$ as divergence between two identical distributions τ and P here would be 0.

■

(20250123#64)

Prove this: Let P be the PMF of a source over a finite alphabet $\mathcal{A}$, and let $x^n \in A_n(\tau)$, i.e., the sequence x^n has type τ. Then:

$$\frac{2^{nH(\tau)}}{(n+1)^{|\mathcal{A}|}} \leq |A_n(\tau)| \leq 2^{nH(\tau)}$$

Proof:

- Consider sequences x^n of type τ, which means each symbol $a \in \mathcal{A}$ appears exactly $n\tau(a)$ times.

- The number of such sequences is given by the multinomial coefficient:

$$|A_n(\tau)| = \frac{n!}{\prod_{a\in\mathcal{A}}(n\tau(a))!}$$

- By Stirling's approximation:

$$m! = m^m e^{-m}\sqrt{2\pi m}(1+o(1)) \quad \text{as } m \to \infty$$

- Taking logarithms and simplifying:

$$\log|A_n(\tau)| = \log n! - \sum_{a\in\mathcal{A}} \log(n\tau(a))! \approx n\log n - n - \sum_{a\in\mathcal{A}} (n\tau(a)\log(n\tau(a)) - n\tau(a))$$

- This simplifies to:

$$\log|A_n(\tau)| \approx -n\sum_{a\in\mathcal{A}} \tau(a)\log\tau(a) = nH(\tau)$$

- Therefore,

$$|A_n(\tau)| \leq 2^{nH(\tau)}$$

  Upperbound also follows using this argument:

$$\begin{aligned} 1 \geq \tau_n(A_n(\tau)) &= \sum_{x^n\in A_n(\tau)} \tau_n(x^n) \\ &= \sum_{x^n\in A_n(\tau)} 2^{-nH(\tau)} \\ &= |A_n(\tau)|2^{-nH(\tau)} \end{aligned}$$

- For the lower bound, note that the number of distinct types τ is at most $(n+1)^{|\mathcal{A}|}$, and the total number of sequences is $|\mathcal{A}|^n$.
- Since $\bigcup_{\tau\in\mathcal{T}_n} A_n(\tau) = \mathcal{A}^n$, there must exist some τ such that:

$$|A_n(\tau)| \geq \frac{|\mathcal{A}|^n}{(n+1)^{|\mathcal{A}|}} = \frac{2^{n\log_2|\mathcal{A}|}}{(n+1)^{|\mathcal{A}|}} \geq \frac{2^{nH(\tau)}}{(n+1)^{|\mathcal{A}|}}$$

- Hence,

$$\frac{2^{nH(\tau)}}{(n+1)^{|\mathcal{A}|}} \leq |A_n(\tau)| \leq 2^{nH(\tau)}$$

■

(20250123#65)

Prove this:

$$\frac{2^{-nD(\tau\|P)}}{(n+1)^{|A|}} \leq P_n(|A_n(\tau)|) \leq 2^{-nD(\tau\|P)}$$

Let P be a probability mass function (PMF) over a finite alphabet $\mathcal{A}$. Let τ be a type (empirical distribution) over $\mathcal{A}$, and let $A_n(\tau)$ denote the set of all sequences $x^n \in \mathcal{A}^n$ of type τ. Then:

$$\frac{2^{-nD(\tau\|P)}}{(n+1)^{|\mathcal{A}|}} \leq P^n(A_n(\tau)) \leq 2^{-nD(\tau\|P)}$$

Proof:

Let $P^n(x^n)$ denote the probability of the sequence x^n under the product distribution induced by P. If $x^n \in A_n(\tau)$, then all such sequences have the same probability:

$$P^n(x^n) = \prod_{i=1}^{n} P(x_i) = \prod_{a\in\mathcal{A}} P(a)^{n\tau(a)} = 2^{-n\sum_{a\in\mathcal{A}}\tau(a)\log\frac{1}{P(a)}} = 2^{-n(H(\tau)+D(\tau\|P))}$$

Therefore,

$$P^n(x^n) = 2^{-n(H(\tau)+D(\tau\|P))} = 2^{-nH(\tau)} \cdot 2^{-nD(\tau\|P)}$$

Now, consider the total probability of the type class:

$$P^n(A_n(\tau)) = |A_n(\tau)| \cdot P^n(x^n) \quad \text{for any } x^n \in A_n(\tau)$$

Using the bounds on the size of the type class from a previous lemma:

$$\frac{2^{nH(\tau)}}{(n+1)^{|\mathcal{A}|}} \leq |A_n(\tau)| \leq 2^{nH(\tau)}$$

Multiplying both sides by $2^{-n(H(\tau)+D(\tau\|P))}$, we obtain:

$$\frac{2^{nH(\tau)}}{(n+1)^{|\mathcal{A}|}} \cdot 2^{-n(H(\tau)+D(\tau\|P))} = \frac{2^{-nD(\tau\|P)}}{(n+1)^{|\mathcal{A}|}} \leq P^n(A_n(\tau)) \leq 2^{nH(\tau)} \cdot 2^{-n(H(\tau)+D(\tau\|P))} = 2^{-nD(\tau\|P)}$$

Hence,

$$\frac{2^{-nD(\tau\|P)}}{(n+1)^{|\mathcal{A}|}} \leq P^n(A_n(\tau)) \leq 2^{-nD(\tau\|P)}$$

■

(20250123#66)

Give a new universal coding scheme based on type classes with bounded empirical entropy:

Let $\mathcal{A}$ be a finite alphabet and consider an i.i.d. source over $\mathcal{A}$ with true PMF P. We define a universal coding scheme based on type classes with bounded empirical entropy.

Definition:

Let $C_n \subset \mathcal{A}^n$ be defined as:

$$C_n := \bigcup_{\tau : H(\tau) \leq r} A_n(\tau)$$

That is, C_n contains all sequences of length n whose empirical distribution τ satisfies $H(\tau) \leq r$.

Encoding Strategy:

- Assign a 1-1 code to the set C_n.
- If $x^n \in C_n$, transmit the index of x^n in a fixed enumeration.
- If $x^n \notin C_n$, transmit a flag indicating error or use a fallback strategy.

Bounding the Size of C_n

The size of the codebook is bounded as:

$$|C_n| = \sum_{\tau : H(\tau) \leq r} |A_n(\tau)| \leq \sum_{\tau : H(\tau) \leq r} 2^{nH(\tau)} \leq 2^{nr} |\mathcal{T}_n|$$

Using the standard bound:

$$|\mathcal{T}_n| \leq (n+1)^{|\mathcal{A}|}$$

we get:

$$|C_n| \leq 2^{nr} (n+1)^{|\mathcal{A}|}$$

Asymptotic Encoding Rate:

Let the length of the binary description be:

$$\ell_n(x^n) = \lceil \log |C_n| \rceil$$

Then the asymptotic rate satisfies:

$$\lim_{n \to \infty} \frac{\ell_n(x^n)}{n} \leq \lim_{n \to \infty} \frac{\log\left(2^{nr}(n+1)^{|\mathcal{A}|}\right) + 1}{n} = r$$

Hence, this coding scheme achieves an encoding rate r, independent of the true PMF P, provided $H(P) < r$.

Error Probability:

Let $e_n(f, \varphi)$ be the probability of error when using encoder f and decoder φ, due to $x^n \notin C_n$. Then:

$$e_n(f, \varphi) = P^n(C_n^c) = \sum_{\tau : H(\tau) > r} P^n(A_n(\tau))$$

Using the type class probability bound:

$$P^n(A_n(\tau)) \leq 2^{-nD(\tau \| P)}$$

and the number of types bounded by $|\mathcal{T}_n| \leq (n+1)^{|\mathcal{A}|}$, we get:

$$e_n(f, \varphi) \leq \sum_{\tau : H(\tau) > r} 2^{-nD(\tau \| P)} \leq (n+1)^{|\mathcal{A}|} \cdot 2^{-nD}$$

where

$$D := \min_{\tau : H(\tau) > r} D(\tau \| P)$$

If $H(P) < r$, then $D > 0$, and the error probability decays exponentially:

$$e_n(f, \varphi) \leq 2^{-nD + |\mathcal{A}| \log(n+1)} \to 0 \quad \text{as } n \to \infty$$

Conclusion:

This scheme is:

- **Universal:** Neither the encoder nor the decoder needs to know the true PMF P.
- **Efficient:** Achieves an encoding rate arbitrarily close to r.
- **Reliable:** Has vanishing error probability when $H(P) < r$.

(20250128#67)

Give a third proof of the source coding theorem:

A Third Proof of the Source Coding Theorem

To introduce *random codes*, we proceed as follows.

- Fix the **rate** R and blocklength n.
- Define $M_n = \lceil 2^{nR} \rceil$ so that M_n is an integer — this is the number of codewords.
- Note that $\frac{\log M_n}{n} \to R$ as $n \to \infty$.
- The set of codeword indices is $\{1, 2, \ldots, M_n\}$.

Random Encoder Construction

- For each $x^n = (x_1, \ldots, x_n) \in \mathcal{A}^n$, select an index i uniformly at random from $\{1, 2, \ldots, M_n\}$.
- Define $f(x^n) = i$. This defines one realization of the encoder f.
- Since f is randomly chosen, it is a random variable. Let F denote the **random code**, where a realization is a function $f : \mathcal{A}^n \to \{1, 2, \ldots, M_n\}$.
- So F takes values in the space of functions from $\mathcal{A}^n$ to $\{1, \ldots, M_n\}$.
- The source string X^n is random with distribution P_{X^n}.

When Does Error Occur?

- Error occurs if two elements $x^n, \tilde{x}^n \in \mathcal{A}^n$ map to the same index i under f, and at least one of them is in the typical set $A(n, \delta)$.

Decoder Construction

- Define the decoder $\phi : \{1, 2, \ldots, M_n\} \to \mathcal{A}^n$ as follows:
  - Let $f^{-1}(i) = \{x^n \in \mathcal{A}^n : f(x^n) = i\}$.
  - If $f^{-1}(i) \cap A(n, \delta)$ is a singleton $\{x^n\}$, then set $\phi(i) = x^n$.
  - Otherwise, set $\phi(i) = a^n$, a fixed reference string in $\mathcal{A}^n$.

Error Event

- An error occurs if $\phi(f(x^n)) \neq x^n$, i.e., the decoded output is not equal to the input.
- There are two sources of randomness:
  1. The input x^n is random according to P_{X^n}.

2. The function f is chosen randomly (random code construction).

- The average error probability is:

$$\mathbb{P}(\text{Error}) = \mathbb{E}_{F,X^n}\left[\mathbf{1}\{\phi(F(X^n)) \neq X^n\}\right] = \sum_f P_F(f) \sum_{x^n} P_{X^n}(x^n)\mathbf{1}\{\phi(f(x^n)) \neq x^n\}$$

Split the error into two parts: whether $X^n \in A(n, \delta)$ (typical set) or not:

$$\mathbb{P}(\mathcal{E}) = \mathbb{P}(X^n \notin A(n,\delta)) + \mathbb{P}(\mathcal{E} \mid X^n \in A(n,\delta)) \cdot \mathbb{P}(X^n \in A(n,\delta))$$

- The first term $\mathbb{P}(X^n \notin A(n,\delta)) \to 0$ as $n \to \infty$ by the Asymptotic Equipartition Property (AEP).
- For the second term, we upper bound the error over the typical set.

Fix a realization of the encoder f. Let $x^n \in A(n,\delta)$ be a typical string. Then, an error occurs only if there exists another $x'^n \neq x^n$ such that:

$$x'^n \in A(n,\delta), \quad f(x'^n) = f(x^n)$$

Since each x^n is mapped independently and uniformly to $\{1, \ldots, M_n\}$, the probability that any fixed $x'^n \in A(n,\delta) \setminus \{x^n\}$ collides with x^n is $\frac{1}{M_n}$. By union bound:

$$\mathbb{P}(\text{collision} \mid x^n \in A(n,\delta)) \leq \frac{|A(n,\delta)| - 1}{M_n} \leq \frac{2^{n(H(P_X)+\delta)}}{2^{nR}}$$

Putting it All Together

Hence, total probability of error:

$$\mathbb{P}(\mathcal{E}) \leq \mathbb{P}(X^n \notin A(n,\delta)) + \frac{2^{n(H(P_X)+\delta)}}{2^{nR}}$$

Choose $R > H(P_X) + \delta$ for some small $\delta > 0$. Then as $n \to \infty$:

$$\mathbb{P}(\mathcal{E}) \to 0$$

Conclusion

For any rate $R > H(P_X)$, we can construct a code of rate R and vanishing error probability. Therefore:

$$\limsup_{n\to\infty} \frac{r(n,\epsilon)}{n} \leq H(P_X)$$

This completes the achievability part of the source coding theorem via random coding.

Concluding Remarks on Random Source Coding

We analyze the average error probability over the space of random codes and show the existence of a good deterministic code.

$$\mathbb{P}(\text{Error}) = \sum_{f} P_F(f) \cdot e(f, \phi) \to 0 \quad \text{as } n \to \infty$$

This implies that for each n, there exists a deterministic code (f_n, ϕ_n) such that:

$$e(f_n, \phi_n) \leq \mathbb{E}_F[e(f, \phi)] \to 0 \quad \text{as } n \to \infty$$

Therefore, there exists a sequence of codes $\{(f_n, \phi_n)\}_{n \geq 1}$ such that:

$$\lim_{n \to \infty} e(f_n, \phi_n) = 0$$

Interpretation

There exists a (deterministic) mapping such that the probability of decoding error vanishes as the length of the sequence to be encoded grows.

Practical Considerations

- The decoder needs knowledge of the source distribution $P(\cdot)$ because the decoding rule depends on the typical set $A(n, \delta)$.
- The encoder also needs knowledge of $P(\cdot)$:
  - To construct the function f_n.
  - To select appropriate values of rate R and slack parameter δ.
- The random coding argument explores the space of all possible encoders and decoders in an indirect manner — this is a form of design space exploration.

Extension to Stationary and Ergodic Sources

The same random coding argument applies to general stationary and ergodic sources, provided we define the typical set $A(n, \delta)$ appropriately for that class of sources.

(20250128#68)
What are canonical codes?

Kraft Inequality and Instantaneous Codes

Let $L(x)$ denote the length of the codeword for symbol $x \in \mathcal{A}$. For instantaneous (prefix-free) codes, the **Kraft inequality** holds:

$$\sum_{x \in \mathcal{A}} 2^{-L(x)} \leq 1$$

Instantaneous codes are those where no codeword is a prefix of another.

Properties of Optimal Codes

- If $P(a_1) > P(a_2)$, then $L(a_1) \leq L(a_2)$.

  Proof (by contradiction): Suppose $L(a_1) > L(a_2)$. Exchanging the two codewords results in a change in expected length:

  $$\Delta = (P(a_1) - P(a_2))(L(a_1) - L(a_2)) > 0$$

  which contradicts optimality. Hence, more probable symbols have shorter codewords.

- The two longest codewords must have the same length.

  Proof: Suppose not. Then the longest codeword is longer than all others. Trim it by one bit to get a shorter codeword, and modify the tree accordingly. This yields a prefix-free code with strictly smaller expected length, contradicting optimality.

- The longest codeword must have a sibling.

  Proof: If a codeword is not part of a sibling pair (i.e., doesn't share its prefix with another codeword), trimming and restructuring again leads to a shorter prefix-free code. Thus, the longest codeword must have a sibling of the same length.

- The two symbols with the lowest probabilities can be made siblings.

  Proof sketch: If the lowest probability symbols are not siblings, restructure the code tree to make them siblings (possibly by exchanging with those that are). This does not increase the expected length and retains optimality.

Codes satisfying the above structural properties are called **canonical codes**.

(20250128#69)

Prove the optimality of Huffman codes:

Let C^* be the Huffman code. Then,

$$\mathbb{E}[L_{C^*}] \leq \min_{C:\text{satisfies Kraft inequality}} \mathbb{E}[L_C]$$

Proof (by Induction on $|\mathcal{A}|$)

- **Base Case**: For $|\mathcal{A}| = 2$, any prefix-free binary code assigns one symbol to 0 and the other to 1. This is optimal since the expected length is exactly 1.
- **Inductive Step**:

  Suppose Huffman coding is optimal for alphabet size $n - 1$. Now consider $\mathcal{A} = \{a_1, a_2, \ldots, a_n\}$ with $P(a_1) \geq P(a_2) \geq \cdots \geq P(a_n)$.

  - Merge the two least probable symbols a_{n-1} and a_n into a new symbol a' with

    $$P(a') = P(a_{n-1}) + P(a_n)$$

    forming a new alphabet $\mathcal{A}'$ with $n - 1$ symbols.
  - Let L'_H be the optimal code lengths for $\mathcal{A}'$ from the inductive hypothesis.
  - Append 0 and 1 to the codeword for a' to obtain codewords for a_{n-1} and a_n.
  - The expected length for the Huffman code L_H becomes:

    $$\mathbb{E}[L_H] = \mathbb{E}[L'_H] + 1 \cdot (P(a_{n-1}) + P(a_n))$$

  - Any optimal code must also assign sibling codewords to a_{n-1} and a_n. If not, rearranging the code tree can produce a better code, contradicting optimality.

  Key Structural Claim:

  Any optimal prefix-free code must assign **sibling codewords** to the two least probable symbols, say a_{n-1} and a_n.

  Justification: Suppose not. That is, in an optimal code tree, a_{n-1} and a_n are not siblings.

  Then, there must exist some other pair of sibling codewords at the maximum depth of the tree, say a_i and a_j (with $P(a_i), P(a_j) > P(a_{n-1}), P(a_n)$).

  Now, exchange the codewords of a_i, a_j with those of a_{n-1}, a_n (keeping lengths unchanged). Since $P(a_i) > P(a_{n-1})$ and $P(a_j) > P(a_n)$, this rearrangement strictly **reduces** the expected length as the least probable codewords now have the most length:

  $$\Delta = (P(a_i) - P(a_{n-1}))(L_i - L_{n-1}) + (P(a_j) - P(a_n))(L_j - L_n) > 0$$

  contradicting the assumption that the original code was optimal.

  Hence, Huffman's construction yields an optimal prefix-free code.

(20250128#70)

Explain how divergence comes into the picture of binary hypothesis testing:

- Let $X_1, X_2, \ldots, X_n$ be i.i.d. random variables taking values in a finite set $\mathcal{A}$.
- We consider a binary hypothesis testing problem:

$$\begin{aligned}
&H_0 : X_1, X_2, \ldots, X_n \sim \text{i.i.d. } P_0 \quad \text{(null hypothesis — normal condition)}\\
&H_1 : X_1, X_2, \ldots, X_n \sim \text{i.i.d. } P_1 \quad \text{(alternative — abnormal or event condition)}
\end{aligned}$$

- The observation space is $\mathcal{A}^n$. Define a decision region $D_0 \subset \mathcal{A}^n$ where we declare H_0.
- The type-I error (false alarm) is defined as:

$$\alpha_n = P_0^{(n)}(D_0^c) \leq \epsilon$$

- The type-II error (missed detection) is:

$$\beta_n = P_1^{(n)}(D_0)$$

- Define:

$$w(n, \epsilon) := \min_{D_0 \subset \mathcal{A}^n : P_0^{(n)}(D_0) \geq 1-\epsilon} P_1^{(n)}(D_0)$$

- Then, from Sanov's theorem (or strong converse of Stein's Lemma), we have:

$$\lim_{n\to\infty} -\frac{1}{n} \log w(n, \epsilon) = D(P_0 \| P_1)$$

- Therefore, under the constraint $\alpha_n \leq \epsilon$, the best achievable type-II error probability β_n satisfies:

$$\beta_n \approx 2^{-nD(P_0\|P_1)}$$

  which decays exponentially fast with n.
- This exponential decay reflects the fact that relative entropy (KL divergence) determines how well we can distinguish P_0 from P_1.
- More advanced techniques like Chernoff bounds or Chernoff entropy allow us to analyze the *simultaneous* exponential decay of both α_n and β_n.

(20250128#71)

Explain method of types:

- Let $\mathcal{A}$ be a finite alphabet.

- Let P_1 be a probability mass function on $\mathcal{A}$.
- Let τ_0 be a **type**, i.e., an empirical distribution of some sequence $x^n \in \mathcal{A}^n$.
- The type τ_0 is a probability mass function on $\mathcal{A}$ such that $n\tau_0(a)$ is an integer for all $a \in \mathcal{A}$.
- The type class corresponding to τ_0 is the set:

  $$A_n(\tau_0) := \{x^n \in \mathcal{A}^n : \tau(x^n) = \tau_0\}$$

  where $\tau(x^n)$ denotes the empirical distribution (type) of x^n.
- The probability of the type class under $P_1^{(n)}$ (i.i.d. with distribution P_1) satisfies:

  $$\frac{1}{(n+1)^{|\mathcal{A}|}} \cdot 2^{-nD(\tau_0\|P_1)} \leq P_1^{(n)}\left(A_n(\tau_0)\right) \leq 2^{-nD(\tau_0\|P_1)}$$

  where $D(\tau_0\|P_1)$ is the Kullback-Leibler divergence between the type τ_0 and distribution P_1.

(20250128#72)

Prove this inequality:

$$D(P_{X^n}\|\prod_{i=1}^{n} Q_{X_i}) \geq \sum_{i=1}^{n} D(P_{X_i}\|Q_{X_i})$$

- **Non-negativity:** For any two probability distributions P and Q on a common finite alphabet $\mathcal{A}$,

  $$D(P\|Q) \geq 0$$

  with equality if and only if $P = Q$. This is analogous to a squared distance but is not a metric.
- **Log-sum Inequality:** For non-negative sequences $\{p_i\}$ and $\{q_i\}$,

  $$\sum_i p_i \log \frac{p_i}{q_i} \geq \left(\sum_i p_i\right) \log \left(\frac{\sum_i p_i}{\sum_i q_i}\right)$$

  Equality holds if and only if $\frac{p_i}{q_i}$ is constant for all i such that $p_i > 0$.
- Let $X^n = (X_1, X_2, \ldots, X_n)$ be a random vector with joint distribution P_{X^n} over $\mathcal{A}^n$.
- Let $Q_{X_1}, Q_{X_2}, \ldots, Q_{X_n}$ be arbitrary marginal distributions on $\mathcal{A}$.

- Define the product distribution

$$Q_{X^n} := \prod_{i=1}^{n} Q_{X_i}$$

  which corresponds to independent random variables with marginals Q_{X_i}.

- Consider the relative entropy between P_{X^n} and Q_{X^n}:

$$D(P_{X^n} \| Q_{X^n}) = \sum_{x^n \in \mathcal{A}^n} P_{X^n}(x^n) \log \frac{P_{X^n}(x^n)}{\prod_{i=1}^{n} Q_{X_i}(x_i)}$$

  This can be rewritten as:

$$D(P_{X^n} \| Q_{X^n}) = \sum_{x^n} P_{X^n}(x^n) \left[\log \frac{P_{X^n}(x^n)}{\prod_{i=1}^{n} P_{X_i}(x_i)} + \log \frac{\prod_{i=1}^{n} P_{X_i}(x_i)}{\prod_{i=1}^{n} Q_{X_i}(x_i)} \right]$$

$$= D(P_{X^n} \| \prod_{i=1}^{n} P_{X_i}) + \sum_{i=1}^{n} D(P_{X_i} \| Q_{X_i})$$

- Hence, we get the inequality:

$$D(P_{X^n} \| \prod_{i=1}^{n} Q_{X_i}) \geq \sum_{i=1}^{n} D(P_{X_i} \| Q_{X_i})$$

  Equality holds if and only if $P_{X^n} = \prod_{i=1}^{n} P_{X_i}$ (i.e., $X_1, \ldots, X_n$ are independent under P).

(20250128#73)

State and give an expression of conditional relative entropy:

Let $P_{X,Y}$ and $Q_{X,Y}$ be two joint distributions on $\mathcal{A} \times \mathcal{B}$, with marginals P_X and Q_X on $\mathcal{A}$, and conditionals $P_{Y|X}(\cdot|x)$ and $Q_{Y|X}(\cdot|x)$ on $\mathcal{B}$ given $x \in \mathcal{A}$.

For a fixed $a \in \mathcal{A}$, the **conditional relative entropy** between $P_{Y|X}(\cdot|a)$ and $Q_{Y|X}(\cdot|a)$ is defined as:

$$D(P_{Y|X}(\cdot|a) \| Q_{Y|X}(\cdot|a)) = \sum_{b \in \mathcal{B}} P_{Y|X}(b|a) \log \frac{P_{Y|X}(b|a)}{Q_{Y|X}(b|a)}$$

Now consider the expected value of this quantity over P_X:

$$\sum_{a \in \mathcal{A}} P_X(a) D(P_{Y|X}(\cdot|a) \| Q_{Y|X}(\cdot|a)) = \sum_{a \in \mathcal{A}} P_X(a) \sum_{b \in \mathcal{B}} P_{Y|X}(b|a) \log \frac{P_{Y|X}(b|a)}{Q_{Y|X}(b|a)}$$

This can be rewritten as:

$$\sum_{a,b} P_X(a)P_{Y|X}(b|a)\log\frac{P_{Y|X}(b|a)}{Q_{Y|X}(b|a)} = \sum_{a,b} P_{X,Y}(a,b)\log\frac{P_{Y|X}(b|a)}{Q_{Y|X}(b|a)}$$

The conditional relative entropy here is represented as

$$D(P_{Y|X}(Y|X)\|Q_{Y|X}(Y|X)|P_X)$$

Thus, the conditional relative entropy is given by the expectation:

$$\mathbb{E}_{P_{X,Y}}\left[\log\frac{P_{Y|X}(Y|X)}{Q_{Y|X}(Y|X)}\right]$$

This expression captures how different the conditionals $P_{Y|X}$ and $Q_{Y|X}$ are, on average, under the joint distribution $P_{X,Y}$.

(20250128#74)

Prove the theorem:

$$D(P_Y\|Q_Y) \leq D(P_{Y|X}\|Q_{Y|X}|P_X)$$

Theorem (Data Processing Inequality for KL Divergence)

Let $P_{X,Y}$ and $Q_{X,Y}$ be joint distributions on $\mathcal{X}\times\mathcal{Y}$ with corresponding marginals P_Y, Q_Y and conditionals $P_{Y|X}, Q_{Y|X}$. Then,

$$D(P_Y\|Q_Y) \leq D(P_{Y|X}\|Q_{Y|X} \mid P_X)$$

Proof:

Recall the definition of conditional KL divergence:

$$D(P_{Y|X}\|Q_{Y|X} \mid P_X) = \sum_{x\in\mathcal{X}} P_X(x)D(P_{Y|X}(\cdot|x)\|Q_{Y|X}(\cdot|x)) = \sum_{x,y} P_{X,Y}(x,y)\log\frac{P_{Y|X}(y|x)}{Q_{Y|X}(y|x)}$$

Also,

$$D(P_Y\|Q_Y) = \sum_y P_Y(y)\log\frac{P_Y(y)}{Q_Y(y)}$$

Now consider the difference:

$$D(P_{Y|X}\|Q_{Y|X} \mid P_X) - D(P_Y\|Q_Y) = \sum_{x,y} P_{X,Y}(x,y)\log\frac{P_{Y|X}(y|x)}{Q_{Y|X}(y|x)} - \sum_y P_Y(y)\log\frac{P_Y(y)}{Q_Y(y)}$$

Combine both terms into a single expectation:

$$= \sum_{x,y} P_{X,Y}(x,y)\left[\log\frac{P_{Y|X}(y|x)}{Q_{Y|X}(y|x)} - \log\frac{P_Y(y)}{Q_Y(y)}\right] = \sum_{x,y} P_{X,Y}(x,y)\log\left(\frac{P_{Y|X}(y|x)/P_Y(y)}{Q_{Y|X}(y|x)/Q_Y(y)}\right)$$

Define the ratio:

$$R(x,y) = \frac{P_{Y|X}(y|x)/P_Y(y)}{Q_{Y|X}(y|x)/Q_Y(y)} \Rightarrow \text{ Then the difference becomes: } \mathbb{E}_{P_{X,Y}}[\log R(X,Y)]$$

Since log is concave and $R(x,y)$ is a likelihood ratio, this expectation is always non-negative (Gibbs' inequality). Therefore:

$$D(P_Y\|Q_Y) \le D(P_{Y|X}\|Q_{Y|X} \mid P_X)$$

Equality Condition:

Equality holds if and only if:

$$\frac{P_{Y|X}(y|x)}{P_Y(y)} = \frac{Q_{Y|X}(y|x)}{Q_Y(y)} \quad \text{for all } x,y \text{ such that } P_{X,Y}(x,y) > 0$$

That is,

$$\frac{P_{Y|X}(y|x)}{Q_{Y|X}(y|x)} = \frac{P_Y(y)}{Q_Y(y)} \quad \forall x,y$$

Which implies that the likelihood ratio between conditionals is independent of x.

(20250128#75)

Is divergence is a convex function? What does it say about conditional KL divergence?

Yes; Let $f(P,Q) := D(P\|Q)$. Then $f(P,Q)$ is a convex function of the pair (P,Q). That is,

$$f(\alpha P_1 + (1-\alpha)P_2, \alpha Q_1 + (1-\alpha)Q_2) \le \alpha f(P_1,Q_1) + (1-\alpha)f(P_2,Q_2)$$

for all $0 \le \alpha \le 1$, and all probability distributions P_1, P_2, Q_1, Q_2 on a finite set.

Interpretation:

Let

$$P = \alpha P_1 + (1-\alpha)P_2, \quad Q = \alpha Q_1 + (1-\alpha)Q_2$$

Then the KL divergence between P and Q is less than or equal to the convex combination of the individual divergences.

Application to Conditional KL Divergence:

Let $P_{Y|X}$ and $Q_{Y|X}$ be conditional distributions. Suppose X takes values in $\{1,2\}$, and let:

$$P_1 = P_{Y|X}(\cdot|1), \quad P_2 = P_{Y|X}(\cdot|2), \quad Q_1 = Q_{Y|X}(\cdot|1), \quad Q_2 = Q_{Y|X}(\cdot|2)$$

Suppose $P_X(1) = \alpha$ and $P_X(2) = 1-\alpha$. Then the conditional relative entropy is:

$$D(P_{Y|X} \| Q_{Y|X} \mid P_X) = \alpha D(P_1 \| Q_1) + (1-\alpha) D(P_2 \| Q_2)$$

This shows that the conditional KL divergence $D(P_{Y|X} \| Q_{Y|X} \mid P_X)$ is a convex combination of pointwise divergences, preserving the convexity structure.

(20250128#76)

Explain sufficient statistics in the context of hypothesis testing:

In the classical statistical setup, hypothesis testing is framed as the problem:

$$\mathcal{T} = \{P_{X|\theta} : \theta \in \mathcal{H}\}$$

We observe a sample X and aim to infer the parameter θ.

A statistic $Y = T(X)$ is said to be **sufficient** for θ if:

$$P_{X|Y,\theta}(x|y,\theta) = P_{X|Y}(x|y)$$

That is, given Y, the distribution of X does not depend on θ anymore. Intuitively, Y captures all the information in X relevant to inferring θ.

This can also be viewed through a Markov chain perspective: if $\theta \to X \to Y$ forms a Markov chain, and Y is sufficient, then $\theta \to Y \to X$ also forms a Markov chain.

(20250128#77)

Consider a Markov chain:

$$X \xrightarrow{P_{Y|X}} Y$$

Let P_X and Q_X be two input distributions, with corresponding output distributions P_Y and Q_Y induced via the same channel $P_{Y|X}$. Then:

Theorem (Data Processing Inequality)**.**

$$D(P_Y \| Q_Y) \leq D(P_X \| Q_X)$$

Prove this theorem:

This is sometimes interpreted as an information-theoretic analog of the second law of thermodynamics: *processing cannot increase information.*

Proof:

We use the chain rule for relative entropy:

$$D(P_X \| Q_X) = D(P_Y \| Q_Y) + D(P_{X|Y} \| Q_{X|Y} \mid P_Y)$$

Explanation:

$$D(P_X \| Q_X) = \mathbb{E}_{P_X}\left[\log \frac{P_X(X)}{Q_X(X)}\right]$$

Using the joint distributions $P_{X,Y} = P_X P_{Y|X}$ and $Q_{X,Y} = Q_X P_{Y|X}$, and marginalizing over Y, we can apply the chain rule:

$$D(P_X \| Q_X) = D(P_Y \| Q_Y) + \mathbb{E}_{P_Y}\left[D(P_{X|Y} \| Q_{X|Y})\right]$$

Since KL divergence is always non-negative, we conclude:

$$D(P_Y \| Q_Y) \leq D(P_X \| Q_X)$$

with equality if and only if $P_{X|Y} = Q_{X|Y}$ almost surely under P_Y.

(20250128#78)

Prove the chain rule of conditional relative entropy:

Let P_{XY} and Q_{XY} be two joint distributions on the same space $\mathcal{X} \times \mathcal{Y}$. Then the chain rule states:

$$D(P_{XY} \| Q_{XY}) = D(P_X \| Q_X) + D(P_{Y|X} \| Q_{Y|X} \mid P_X)$$

where

$$D(P_{Y|X} \| Q_{Y|X} \mid P_X) = \sum_{x \in \mathcal{X}} P_X(x) D(P_{Y|X}(\cdot|x) \| Q_{Y|X}(\cdot|x))$$

Proof:

By the definition of relative entropy,

$$D(P_{XY}\|Q_{XY}) = \sum_{x,y} P_{XY}(x,y)\log\frac{P_{XY}(x,y)}{Q_{XY}(x,y)}$$

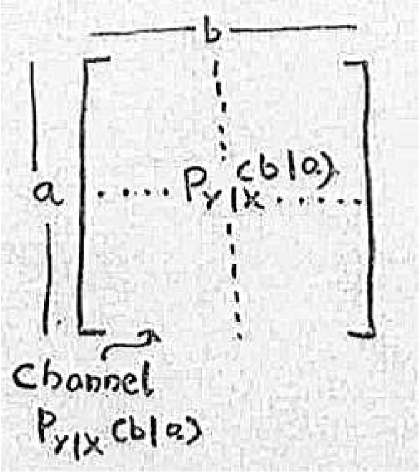
We write both P_{XY} and Q_{XY} in terms of marginals and conditionals:

$$\begin{aligned}
&= \sum_{x,y} P_{XY}(x,y)\log\left(\frac{P_X(x)P_{Y|X}(y|x)}{Q_X(x)Q_{Y|X}(y|x)}\right)\\
&= \sum_{x,y} P_{XY}(x,y)\left[\log\frac{P_X(x)}{Q_X(x)} + \log\frac{P_{Y|X}(y|x)}{Q_{Y|X}(y|x)}\right]\\
&= \sum_{x,y} P_{XY}(x,y)\log\frac{P_X(x)}{Q_X(x)} + \sum_{x,y} P_{XY}(x,y)\log\frac{P_{Y|X}(y|x)}{Q_{Y|X}(y|x)}\\
&= \sum_{x} P_X(x)\log\frac{P_X(x)}{Q_X(x)} + \sum_{x} P_X(x)\sum_{y} P_{Y|X}(y|x)\log\frac{P_{Y|X}(y|x)}{Q_{Y|X}(y|x)}\\
&= D(P_X\|Q_X) + \sum_{x} P_X(x)D(P_{Y|X}(\cdot|x)\|Q_{Y|X}(\cdot|x))\\
&= D(P_X\|Q_X) + D(P_{Y|X}\|Q_{Y|X} \mid P_X)
\end{aligned}$$

■

(20250204#79)

Explain how channels are different from Markov structures:



In the context of channels, unlike Markov processes, the input and output alphabets can differ. That is, for a channel, $X \in \mathcal{X}$, $Y \in \mathcal{Y}$, where $\mathcal{X} \neq \mathcal{Y}$ in general.

- Let P_X be a distribution on $\mathcal{X}$, and $P_{Y|X}$ be the channel transition probability.
- The joint distribution is given by:

$$P_{X,Y} = P_X(x) P_{Y|X}(y|x)$$

- Similarly, let $Q_{Y|X}$ be a second channel. Then, for the same input distribution P_X, the corresponding joint distribution is:

$$Q_{X,Y} = P_X(x) Q_{Y|X}(y|x)$$

Markov Chain Interpretation

The Markov structure induced by the channel is:

$$P_X \to P_{Y|X} \to P_{X,Y}$$

$$P_X \to Q_{Y|X} \to Q_{X,Y}$$

Theorem (Data Processing Inequality)

Let $P_{X,Y} = P_X P_{Y|X}$ and $Q_{X,Y} = P_X Q_{Y|X}$. Then:

$$D(P_Y \| Q_Y) \leq D(P_{X,Y} \| Q_{X,Y}) = D(P_{Y|X} \| Q_{Y|X} \mid P_X)$$

This inequality follows from the fact that marginalization cannot increase divergence — an instance of the data processing inequality for relative entropy.

(20250204#80)

When does the channel mapping become deterministic?

Consider a random variable X passed through a channel defined by a conditional distribution $P_{Y|X}$, resulting in an output random variable Y with marginal distribution P_Y. This can be expressed as:

$$X \xrightarrow{P_{Y|X}} Y \quad \text{with} \quad P_Y(y) = \sum_x P_X(x)P_{Y|X}(y|x)$$

- The mapping $P_{Y|X}$ is a general probabilistic transition rule. It is more general than a deterministic function.
- A special case is when the channel is deterministic. This happens if for every $x \in \mathcal{X}$, there exists a unique $y \in \mathcal{Y}$ such that:

  $$P_{Y|X}(y|x) = \delta_{y=f(x)}$$

  where δ is the Kronecker delta function. In matrix form, this corresponds to each row of the transition matrix $P_{Y|X}$ having a single entry equal to 1, and all others equal to 0.

(20250204#81)

In the theorem:

$$D(P_Y \| Q_Y) \leq D(P_X \| Q_X)$$

across a channel, it says that passing through a channel has resulted in the decrease of the ability to distinguish between P and Q. But when does this inequality become an equality? Meaning, when does passing through a channel keep the distinguishability of the input and output distributions to be same?

Data Processing Inequality (DPI): Equality Condition

Theorem (DPI): Let P_X, Q_X be probability distributions on $\mathcal{X}$, and let $P_{Y|X}$ be a channel from $\mathcal{X}$ to $\mathcal{Y}$. Define:

$$P_Y(y) = \sum_x P_X(x)P_{Y|X}(y|x), \quad Q_Y(y) = \sum_x Q_X(x)P_{Y|X}(y|x)$$

Then,

$$D(P_Y \| Q_Y) \leq D(P_X \| Q_X)$$

Equality Condition: The inequality becomes an equality if and only if the channel $P_{Y|X}$ is *sufficiently informative* about X, in the sense that the conditional distributions $P_{X|Y}$ and $Q_{X|Y}$ are equal P_Y-almost surely:

$$P_{X|Y}(x|y) = Q_{X|Y}(x|y) \quad \text{for all } x, y \text{ with } P_Y(y) > 0$$

Proof:

We begin with the chain rule of relative entropy:

$$D(P_X \| Q_X) = D(P_Y \| Q_Y) + D(P_{X|Y} \| Q_{X|Y} \mid P_Y)$$

This is derived by computing:

$$\begin{aligned} D(P_X \| Q_X) &= \sum_x P_X(x) \log \frac{P_X(x)}{Q_X(x)} \\ &= \sum_{x,y} P_{X,Y}(x,y) \log \frac{P_X(x)}{Q_X(x)} \\ &= \sum_{x,y} P_{X,Y}(x,y) \log \frac{P_{X|Y}(x|y) P_Y(y)}{Q_{X|Y}(x|y) Q_Y(y)} \\ &= \sum_{x,y} P_{X,Y}(x,y) \log \frac{P_Y(y)}{Q_Y(y)} + \sum_{x,y} P_{X,Y}(x,y) \log \frac{P_{X|Y}(x|y)}{Q_{X|Y}(x|y)} \\ &= D(P_Y \| Q_Y) + D(P_{X|Y} \| Q_{X|Y} \mid P_Y) \end{aligned}$$

Since $D(P_{X|Y} \| Q_{X|Y} \mid P_Y) \geq 0$, it follows that:

$$D(P_Y \| Q_Y) \leq D(P_X \| Q_X)$$

Equality Condition: The inequality becomes an equality if and only if:

$$D(P_{X|Y} \| Q_{X|Y} \mid P_Y) = 0$$

which holds if and only if:

$$P_{X|Y}(x|y) = Q_{X|Y}(x|y) \quad \text{for all } x, y \text{ such that } P_Y(y) > 0$$

Interpretation: This condition means that the posterior distributions over X given Y under both models must match. In other words, the observation Y retains all the information about the difference between P_X and Q_X.

(20250204#82)

Give an intuition for second law of thermodynamics using Markov chains and KL divergences:

Let X_0 represent the initial position (state) of molecules in a box, distributed according to $\mathcal{V}_0$:

$$X_0 \sim \nu_0$$

Suppose the system evolves stochastically according to a Markov chain:

$$X_0, X_1, X_2, \ldots$$

with a fixed transition matrix $P = P_{Y|X}$. Let:

$$\nu_k = \text{distribution of } X_k \text{ starting from } \nu_0 \quad \text{and} \quad \nu'_k = \text{distribution starting from } \nu'_0$$

Then:

$$\nu_k = P\nu_{k-1}, \qquad \nu'_k = P\nu'_{k-1}$$

Monotonicity of Relative Entropy:

$$D(\nu_0\|\nu'_0) \geq D(\nu_1\|\nu'_1) \geq D(\nu_2\|\nu'_2) \geq \ldots$$

This reflects the information loss (i.e., distinguishability reduction) as the system evolves. Relative entropy decreases under the action of a stochastic map.

Convergence to Stationary Distribution: As $k \to \infty$, we approach the stationary distribution Π, defined by:

$$\Pi = \Pi P$$

Then:

$$D(\nu_k\|\Pi) \geq D(\nu_{k+1}\|\Pi) \geq \ldots$$

That is, $D(\nu_k\|\Pi)$ is a **Lyapunov function** for the Markovian dynamics — it always decreases over time, although not necessarily to zero.

Interpretation: Relative entropy can be thought of like an energy function that decays and stabilizes at equilibrium. When the system reaches equilibrium, its distribution is the stationary distribution Π.

Special Case — Doubly Stochastic Matrix:

Suppose P is **doubly stochastic** (i.e., both rows and columns sum to 1). Then the uniform distribution:

$$\Pi = \left[\frac{1}{|\mathcal{A}|}, \frac{1}{|\mathcal{A}|}, \ldots\right]$$

is stationary:

$$\Pi P = \Pi$$

Now consider the relative entropy to Π:

$$D(\nu_k\|\Pi) = \sum_{a\in\mathcal{A}} \nu_k(a) \log\left(\frac{\nu_k(a)}{1/|\mathcal{A}|}\right) = \log|\mathcal{A}| - H(X_k)$$

Since $D(\nu_k\|\Pi)$ decreases:

$$\log|\mathcal{A}| - H(X_k) \text{ decreases } \Rightarrow H(X_k) \text{ increases}$$

Therefore:

$$H(X_k) \uparrow \log|\mathcal{A}| \Rightarrow \text{system entropy increases with time}$$

Conclusion — Second Law of Thermodynamics: As the Markov chain evolves:

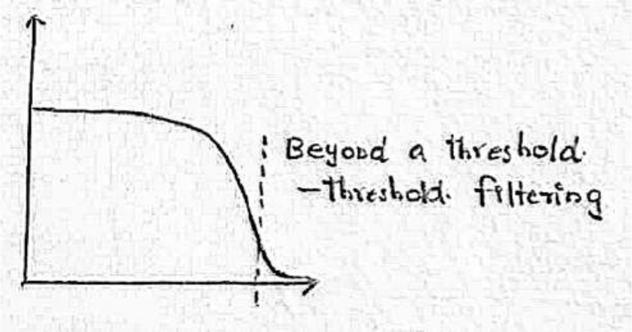
- The system becomes more disordered.
- Entropy increases.
- The distribution tends toward the uniform distribution Π.

This provides an information-theoretic interpretation of the second law of thermodynamics.

(20250204#83)

Describe the threshold filtering phenomenon of shuffling of a deck of cards:

When a deck of cards is shuffled repeatedly using a randomizing process (e.g., riffle shuffle), the distribution of the deck's state evolves over time. Initially, the distribution is far from uniform, but as shuffling continues, the distribution approaches the uniform distribution.



An important phenomenon observed in such processes is:

Threshold Phenomenon (Cutoff Behavior):

- After a certain number of shuffles (the *cutoff threshold*), the distribution suddenly appears very close to uniform.
- Before this threshold, the distribution remains far from uniform.
- This sharp transition is known as **threshold filtering**.

Mathematically, let P_k be the distribution after k shuffles, and let U be the uniform distribution. Then the total variation distance:

$$\delta_k := \frac{1}{2} \sum_{\sigma \in S_n} |P_k(\sigma) - U(\sigma)|$$

drops sharply near the threshold value of k. For example, for a standard 52-card deck under the Gilbert–Shannon–Reeds (GSR) model of riffle shuffling, the threshold is around 7 shuffles.

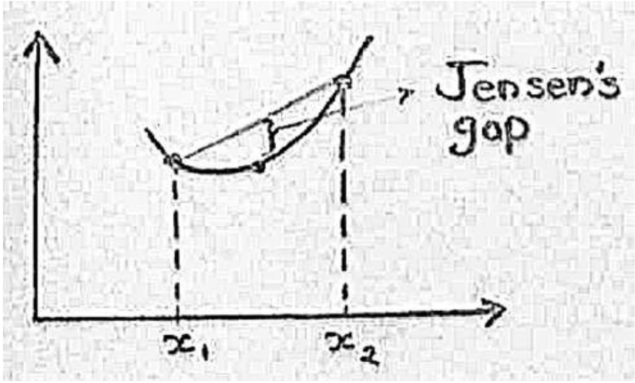
This phenomenon is an example of **filtering by a threshold** — indistinguishability from uniform is achieved sharply and not gradually.

(20250204#84)

State Jensen's inequality. Give its geometric intuition and prove the inequality:

Geometric intuition:

For a convex function f, the graph of f always lies below the chord connecting any two points on the function. This geometric property underlies Jensen's inequality.



Theorem (Jensen's Inequality): Let $f : \mathbb{R}^n \to \mathbb{R}$ be a convex function, and let X be a random variable taking values in $\mathbb{R}^n$. Then:

$$\mathbb{E}[f(X)] \geq f(\mathbb{E}[X])$$

Moreover, if f is strictly convex, then equality holds if and only if $X = \mathbb{E}[X]$ almost surely (i.e., with probability 1).

Proof (Sketch for Discrete Random Variables):

We prove by induction on the support size $|A|$ of X.

- Base case:

  $|A| = 2$, i.e., $X \in \{x_1, x_2\}$ with probabilities $p_1, 1 - p_1$. Then:

  $$\mathbb{E}[f(X)] = p_1 f(x_1) + (1 - p_1)f(x_2) \geq f(p_1 x_1 + (1 - p_1)x_2) = f(\mathbb{E}[X])$$

  This follows directly from the definition of convexity. Equality holds if $x_1 = x_2$, or f is linear (i.e., not strictly convex).

- Inductive Step:

  Assume the result holds for $|A| = n$. Now consider $|A| = n+1$, with values $x_1, \ldots, x_{n+1}$, and associated probabilities $p_1, \ldots, p_{n+1}$. Define:

  $$q = 1 - p_1, \quad \text{and define a new distribution on } x_2, \ldots, x_{n+1} \text{ as } p_i' = \frac{p_i}{q}$$

  Then, by convexity:

  $$\mathbb{E}[f(X)] = p_1 f(x_1) + q \sum_{i=2}^{n+1} p_i' f(x_i) \geq p_1 f(x_1) + q f\left(\sum_{i=2}^{n+1} p_i' x_i\right)$$

  By the base case (or induction), the right-hand side is greater than or equal to:

  $$f\left(p_1 x_1 + q \sum_{i=2}^{n+1} p_i' x_i\right) = f(\mathbb{E}[X])$$

Equality Condition (Strict Convexity Case):

1. All x_i must be equal: $x_1 = x_2 = \cdots = x_{n+1}$, or
2. The function f must be affine (i.e., not strictly convex).

Thus, for strictly convex f, equality holds if and only if $X = \mathbb{E}[X]$ with probability 1.

(20250204#85)

Apply Jensen's inequality to prove non-negativity of divergence:

$$D(P\|Q) = \sum_{a \in \mathcal{A}} P(a) \log \frac{P(a)}{Q(a)} \geq 0$$

with equality if and only if $P = Q$ on the support of P.

We explore this using expectation:

$$\begin{aligned} D(P_X\|Q_X) &= \mathbb{E}_{P_X}\left[\log \frac{P_X(X)}{Q_X(X)}\right] \\ &= -\mathbb{E}_{P_X}\left[\log \frac{Q_X(X)}{P_X(X)}\right] \\ &= \mathbb{E}_{P_X}\left[-\log\left(\frac{Q_X(X)}{P_X(X)}\right)\right] \end{aligned}$$

By Jensen's inequality (since $-\log$ is convex), we have:

$$\mathbb{E}_{P_X}\left[-\log\left(\frac{Q_X(X)}{P_X(X)}\right)\right] \geq -\log\left(\mathbb{E}_{P_X}\left[\frac{Q_X(X)}{P_X(X)}\right]\right)$$

Now compute the inner expectation:

$$\begin{aligned}\mathbb{E}_{P_X}\left[\frac{Q_X(X)}{P_X(X)}\right] &= \sum_{a\in\mathcal{A}} P_X(a)\cdot\frac{Q_X(a)}{P_X(a)} \\ &= \sum_{a\in\mathcal{A}} Q_X(a)\end{aligned}$$

Since this sum is over the support of P_X, and assuming $\text{supp}(P_X) \subseteq \text{supp}(Q_X)$, we get:

$$\sum_{a\in\text{supp}(P_X)} Q_X(a) \leq 1$$

Therefore:

$$-\log\left(\sum_{a\in\text{supp}(P_X)} Q_X(a)\right) \geq 0$$

Which implies:

$$D(P_X\|Q_X) \geq 0$$

(20250208#86)

Define f-divergence:

Let $P = \{p(x)\}$ and $Q = \{q(x)\}$ be two probability mass functions over a discrete alphabet $\mathcal{X}$, and let $f : (0, \infty) \to \mathbb{R}$ be a convex function with $f(1) = 0$. The **discrete f-divergence** from P to Q is defined as:

$$D_f(P\|Q) = \mathbb{E}_Q\left[f\left(\frac{P(X)}{Q(X)}\right)\right] = \sum_{x\in\mathcal{X}} q(x)\, f\left(\frac{p(x)}{q(x)}\right),$$

with the convention that $f\left(\frac{p(x)}{q(x)}\right) = 0$ if $p(x) = q(x) = 0$, and the term is treated as ∞ if $p(x) > 0$ but $q(x) = 0$.

(20250208#87)

Give some examples for f-divergence:

---

1. **Kullback–Leibler (KL) divergence**:

$$f(t) = t\log t, \quad D_{\mathrm{KL}}(P\|Q) = \sum_{x\in\mathcal{X}} p(x)\log\left(\frac{p(x)}{q(x)}\right)$$

2. **Reverse KL divergence**:

$$f(t) = -\log t, \quad D_{\mathrm{KL}}(Q\|P) = \sum_{x\in\mathcal{X}} q(x)\log\left(\frac{q(x)}{p(x)}\right)$$

3. **Total Variation (TV) distance**:

$$f(t) = \frac{1}{2}|t-1|, \quad D_{\mathrm{TV}}(P, Q) = \frac{1}{2}\sum_{x\in\mathcal{X}} |p(x) - q(x)|$$

4. **Chi-squared divergence**:

$$f(t) = (t-1)^2, \quad D_{\chi^2}(P\|Q) = \sum_{x\in\mathcal{X}} \frac{(p(x) - q(x))^2}{q(x)}$$

5. **Hellinger distance**:

$$f(t) = (\sqrt{t} - 1)^2, \quad D_H^2(P, Q) = \sum_{x\in\mathcal{X}} \left(\sqrt{p(x)} - \sqrt{q(x)}\right)^2$$

6. **Jensen–Shannon divergence (symmetric)**:

$$f(t) = t \log t - (t+1)\log\left(\frac{1+t}{2}\right) + \log 2, \quad D_{\mathrm{JS}}(P\|Q) = \frac{1}{2}D_{\mathrm{KL}}(P\|M) + \frac{1}{2}D_{\mathrm{KL}}(Q\|M)$$

where $M = \frac{1}{2}(P+Q)$.

(20250208#88)

Prove the non-negativity of f-divergence:

Theorem (f-divergence non-negativity)**.** *Let $P = \{p(x)\}$ and $Q = \{q(x)\}$ be two probability distributions over a finite set $\mathcal{X}$, and let $f : (0,\infty) \to \mathbb{R}$ be a convex function with $f(1) = 0$. Then,*

$$D_f(P\|Q) = \sum_{x\in\mathcal{X}} q(x) f\left(\frac{p(x)}{q(x)}\right) \geq 0,$$

with equality if and only if $P = Q$ (i.e., $p(x) = q(x)$ for all $x \in \mathcal{X}$).

Proof. The proof relies on **Jensen's inequality** and the convexity of f.

Define the random variable $Z(x) = \frac{p(x)}{q(x)}$ for all $x \in \mathcal{X}$ such that $q(x) > 0$. Note that since P and Q are both probability distributions,

$$\sum_{x\in\mathcal{X}} q(x) \cdot Z(x) = \sum_{x\in\mathcal{X}} p(x) = 1.$$

Applying Jensen's inequality to the convex function f, we get:

$$\sum_{x\in\mathcal{X}} q(x) f(Z(x)) \geq f\left(\sum_{x\in\mathcal{X}} q(x) Z(x)\right) = f(1) = 0.$$

Therefore,

$$D_f(P\|Q) = \sum_{x\in\mathcal{X}} q(x) f\left(\frac{p(x)}{q(x)}\right) \geq 0.$$

Equality holds in Jensen's inequality if and only if $Z(x) = 1$ for all $x \in \mathcal{X}$ with $q(x) > 0$, i.e., $p(x) = q(x)$. Thus, $D_f(P\|Q) = 0 \iff P = Q$. □

(20250208#89)

What is Jeffrey's divergence?

Let $P = \{p(x)\}$ and $Q = \{q(x)\}$ be two discrete probability distributions over a common finite set $\mathcal{X}$. The **Jeffreys divergence** is defined as:

$$J(P\|Q) = D_{\mathrm{KL}}(P\|Q) + D_{\mathrm{KL}}(Q\|P)$$
$$= \sum_{x\in\mathcal{X}} (p(x) - q(x)) \log\left(\frac{p(x)}{q(x)}\right)$$

Properties:

- $J(P\|Q) \geq 0$, with equality if and only if $P = Q$
- Symmetric: $J(P\|Q) = J(Q\|P)$
- Not a true metric (does not satisfy triangle inequality)
- Belongs to the family of f-divergences with $f(t) = (t-1)\log t$

(20250208#90)

Prove this:

$$D_f\left(P_x\|Q_X\right) \geq D_f\left(P_Y\|Q_Y\right)$$

after, let's say, sending through a channel $(A, B, P_{Y|X})$

Data Processing Inequality for f-divergence

Let P_X and Q_X be two probability distributions over a finite set $\mathcal{X}$, and let $P_{Y|X}$ be a stochastic kernel (channel) from X to Y. Define the output distributions:

$$P_Y(y) = \sum_{x\in\mathcal{X}} P_X(x)P_{Y|X}(y|x), \quad Q_Y(y) = \sum_{x\in\mathcal{X}} Q_X(x)P_{Y|X}(y|x)$$

Let $f : (0,\infty) \to \mathbb{R}$ be a convex function with $f(1) = 0$. The f-divergence between two distributions is defined as:

$$D_f(P\|Q) = \sum_{x\in\mathcal{X}} Q(x) f\left(\frac{P(x)}{Q(x)}\right)$$

We aim to show that:

$$D_f(P_X\|Q_X) \geq D_f(P_Y\|Q_Y)$$

Proof

The f-divergence between the output distributions is:

$$D_f(P_Y\|Q_Y) = \sum_{y\in\mathcal{Y}} Q_Y(y) f\left(\frac{P_Y(y)}{Q_Y(y)}\right)$$

Now, observe that:

$$\frac{P_Y(y)}{Q_Y(y)} = \frac{\sum_{x\in\mathcal{X}} P_X(x)P_{Y|X}(y|x)}{\sum_{x\in\mathcal{X}} Q_X(x)P_{Y|X}(y|x)}$$

This is a convex combination of the ratios $\frac{P_X(x)}{Q_X(x)}$ with weights $\frac{Q_X(x)P_{Y|X}(y|x)}{Q_Y(y)}$. Using Jensen's inequality (since f is convex), we get:

$$f\left(\frac{P_Y(y)}{Q_Y(y)}\right) \le \sum_{x\in\mathcal{X}} \frac{Q_X(x)P_{Y|X}(y|x)}{Q_Y(y)} f\left(\frac{P_X(x)}{Q_X(x)}\right)$$

Multiplying both sides by $Q_Y(y)$ and summing over $y \in \mathcal{Y}$, we obtain:

$$\begin{aligned}
D_f(P_Y\|Q_Y) &= \sum_y Q_Y(y) f\left(\frac{P_Y(y)}{Q_Y(y)}\right) \\
&\le \sum_y \sum_x Q_X(x)P_{Y|X}(y|x) f\left(\frac{P_X(x)}{Q_X(x)}\right) \\
&= \sum_x Q_X(x) f\left(\frac{P_X(x)}{Q_X(x)}\right) \sum_y P_{Y|X}(y|x) \\
&= \sum_x Q_X(x) f\left(\frac{P_X(x)}{Q_X(x)}\right) = D_f(P_X\|Q_X)
\end{aligned}$$

Conclusion

Thus, passing the distributions through a channel cannot increase their f-divergence:

$$D_f(P_X\|Q_X) \ge D_f(P_Y\|Q_Y)$$

(20250208#91)

Explain the convex combination of ratios $P_X(x)/Q_X(x)$ step in the previous question's answer.

We aim to understand the convex combination step in the proof of the data processing inequality for f-divergence.

Recall the definition of the output distributions induced by the channel $P_{Y|X}$:

$$P_Y(y) = \sum_x P_X(x)P_{Y|X}(y|x), \quad Q_Y(y) = \sum_x Q_X(x)P_{Y|X}(y|x)$$

Thus, the ratio of output probabilities is:

$$\frac{P_Y(y)}{Q_Y(y)} = \frac{\sum_x P_X(x)P_{Y|X}(y|x)}{\sum_x Q_X(x)P_{Y|X}(y|x)}$$

Now define the weights:

$$w_x(y) = \frac{Q_X(x)P_{Y|X}(y|x)}{Q_Y(y)} \quad \text{so that} \quad \sum_x w_x(y) = 1$$

Therefore, the ratio can be expressed as:

$$\frac{P_Y(y)}{Q_Y(y)} = \sum_x w_x(y) \cdot \frac{P_X(x)}{Q_X(x)}$$

This shows that $\frac{P_Y(y)}{Q_Y(y)}$ is a convex combination of the values $\frac{P_X(x)}{Q_X(x)}$, with the weights $w_x(y)$ depending on the channel and the distribution Q_X.

Since f is convex, by Jensen's inequality:

$$f\left(\frac{P_Y(y)}{Q_Y(y)}\right) = f\left(\sum_x w_x(y) \cdot \frac{P_X(x)}{Q_X(x)}\right) \leq \sum_x w_x(y) \cdot f\left(\frac{P_X(x)}{Q_X(x)}\right)$$

Multiplying both sides by $Q_Y(y)$, and summing over y, we get:

$$\sum_y Q_Y(y) f\left(\frac{P_Y(y)}{Q_Y(y)}\right) \leq \sum_y \sum_x Q_X(x)P_{Y|X}(y|x) f\left(\frac{P_X(x)}{Q_X(x)}\right) = \sum_x Q_X(x) f\left(\frac{P_X(x)}{Q_X(x)}\right)$$

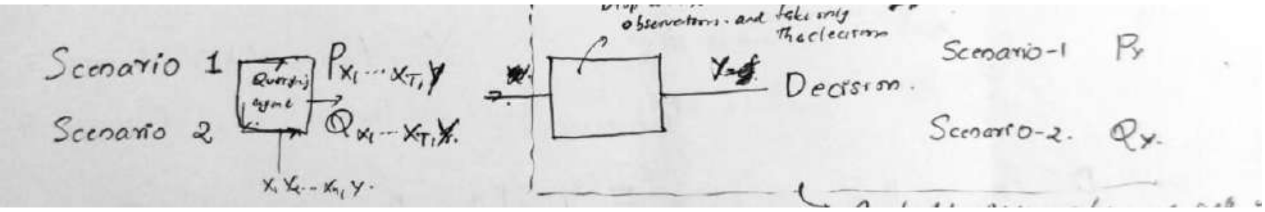
Hence, we obtain the data processing inequality:

$$D_f(P_Y \| Q_Y) \leq D_f(P_X \| Q_X)$$

(20250208#92)

Show that in hypothesis testing the number of observations (T) being user's choice, T will increase with the reduction in error probability.

We are interested in bounding the number of queries or steps T required to achieve an error probability no larger than δ. Let the observations correspond to $X_1, X_2, \ldots, X_T$. We are ob-



serving till we are confident enough to make a decision. $X_1, X_2, \ldots, X_T$ are like the questions asked and Y is the declaration of the decision based on the answers to the questions.

For the two scenarios 1 and 2, let the probabilities be $(\geq 1-\delta), (\leq \delta)$ and $(\leq \delta, \geq 1-\delta)$ respectively.

This derivation typically appears in binary hypothesis testing, where we distinguish between two hypotheses H_0 and H_1 using repeated queries with outcomes modeled as random variables.

Assume that under each hypothesis, the observations $X_1, \ldots, X_T$ are i.i.d. random variables distributed according to P_X under H_0, and Q_X under H_1. Let $D = D(P_{X^n,Y} \| Q_{X^n,Y})$ denote the Kullback-Leibler divergence between the distributions.

$$D(P_Y \| Q_Y) = (1-\delta)\log\left(\frac{1-\delta}{\delta}\right) + \delta \log\left(\frac{\delta}{1-\delta}\right)$$

As δ becomes smaller and smaller, $D(P_Y \| Q_Y) \to \log(1/\delta)$. Total number of queries will have this form:

$$T \times D \geq D(P_Y || Q_Y)$$

which gives

$$T \geq \frac{\log(1/\delta)}{D(P_1 \| P_0)}$$

Use of Chernoff Bound or Sanov's Theorem

The probability of error decays exponentially with the number of samples, i.e.,

$$P_e^{(T)} \leq 2^{-TD}$$

This implies that to ensure an error probability at most δ, we must satisfy:

$$2^{-TD} \leq \delta$$

Take logarithm

$$-TD \leq \log \delta \Rightarrow T \geq \frac{\log(1/\delta)}{D}$$

Conclusion:

Thus, to achieve an error probability of at most δ, the number of steps T must satisfy:

$$T \geq \frac{\log(1/\delta)}{D(P_1 \| P_0)}$$

This bound is fundamental in information-theoretic limits of hypothesis testing and adaptive decision-making. It shows that to get higher accuracy (smaller δ), one needs more queries, inversely proportional to the information gain D per query.

(20250208#93)

Explain in the context of MNIST example of distinguishing between a 5 and a 7:

$$-TD \leq \log \delta \Rightarrow T \geq \frac{\log(1/\delta)}{D}$$

We want to distinguish between two digits from the MNIST dataset: **5** and **7**. This is a binary classification problem and can be framed as a binary hypothesis testing problem:

- H_0: the image is a **5**
- H_1: the image is a **7**

Suppose we observe the image one pixel (or a small region) at a time. Let $X_1, X_2, \ldots, X_T$ denote these observations, and let the final decision (after seeing T pixels) be denoted by the variable Y, where $Y \in \{0, 1\}$ indicates whether we declare the digit to be a 5 or a 7.

The process proceeds as follows:

- At each step, we reveal a small piece of the image (a query or observation).
- After T such observations, we must decide whether the digit is a 5 or a 7.
- Our goal is to make this decision with high confidence — that is, with error probability at most δ.

Let the conditional distributions of observations under each hypothesis be:

$$\text{Under } H_0: \quad X_i \sim P_X, \quad \text{and under } H_1: \quad X_i \sim Q_X$$

We assume $X_1, \ldots, X_T$ are i.i.d. samples from these respective distributions. The KL divergence between these distributions is:

$$D = D(P_X \| Q_X)$$

The probability of correct classification under H_0 is at least $1-\delta$, and under H_1 is at most δ, and vice versa for the second case. So the distributions of the decision variable Y under H_0 and H_1 differ in total variation by approximately $1-2\delta$, and the KL divergence between P_Y and Q_Y is given by:

$$D(P_Y \| Q_Y) = (1-\delta)\log\left(\frac{1-\delta}{\delta}\right) + \delta\log\left(\frac{\delta}{1-\delta}\right)$$

As $\delta \to 0$, we can approximate:

$$D(P_Y \| Q_Y) \approx \log\left(\frac{1}{\delta}\right)$$

$$\begin{aligned}
D(P_{X_1,\ldots,X_T} \| Q_{X_1,\ldots,X_T}) &= \mathbb{E}_{P_{X_1,\ldots,X_T}}\left[\log\left(\frac{P_{X_1,\ldots,X_T}(X_1,\ldots,X_T)}{Q_{X_1,\ldots,X_T}(X_1,\ldots,X_T)}\right)\right] \\
&= \mathbb{E}_{P_{X_1,\ldots,X_T}}\left[\sum_{n=1}^{T}\log\frac{P_{X_n|X_1,\ldots,X_{n-1}}(X_n|X_{1,},\ldots,X_{n-1})}{Q_{X_n|X_1,\ldots,X_{n-1}}(X_n|X_{1,},\ldots,X_{n-1})}\right] \\
&= \mathbb{E}_{P_{X_1,\ldots,X_T}}\left[\sum_{n=1}^{T}\log\frac{P_{X_n}(X_n)}{Q_{X_n}(X_n)}\right] \\
&= \mathbb{E}[T] \times \mathbb{E}_{P_{X_1,\ldots,X_T}}\left[\log\frac{P_{X_1}(X_1)}{Q_{X_1}(X_1)}\right]
\end{aligned}$$

where T is random time, $\mathbb{E}[T]$ is the expected number of queries and the second term of product on RHS is the bottleneck per query.

{ Theorem from Walt'z optimal stopping time }

Information-Theoretic Bound on Number of Queries

The data processing inequality tells us:

$$T \cdot D(P_X \| Q_X) \geq D(P_Y \| Q_Y)$$

Substituting the asymptotic form of $D(P_Y \| Q_Y)$, we get:

$$T \cdot D \geq \log\left(\frac{1}{\delta}\right) \Rightarrow T \geq \frac{\log(1/\delta)}{D(P_X \| Q_X)}$$

Interpretation

This means that in order to reduce the probability of classification error to at most δ, the number of observations (pixels or features) we need to examine must grow at least like $\log(1/\delta)$.

If the KL divergence between the distributions of pixels for a 5 and a 7 is large (i.e., the digits are easy to distinguish), then fewer queries are needed. Otherwise, more evidence (larger T) is required to confidently decide between the two.

(20250211#94)

Explain f-divergence along its definition, properties, and invariance under linear transformations of the convex function:

Let P and Q be probability distributions over the same space, and suppose $P \ll Q$ (i.e., P is absolutely continuous with respect to Q). Then the f-divergence between P and Q, associated with a convex function $f : (0, \infty) \to \mathbb{R}$, is defined as:

$$D_f(P\|Q) = \mathbb{E}_Q\left[f\left(\frac{dP}{dQ}\right)\right] = \sum_x Q(x) f\left(\frac{P(x)}{Q(x)}\right) \quad \text{(in the discrete case)}$$

Key Properties:

1. **Normalization:** If $f(1) = 0$, then $D_f(P\|Q) = 0$ when $P = Q$. This follows from Jensen's inequality:

$$D_f(P\|Q) = \mathbb{E}_Q\left[f\left(\frac{P(x)}{Q(x)}\right)\right] \geq f\left(\mathbb{E}_Q\left[\frac{P(x)}{Q(x)}\right]\right) = f(1)$$

with equality when $P = Q$, hence $D_f(P\|Q) = f(1) = 0$ in that case.

2. **Equivalence under Linear Adjustment:** Suppose we construct a new function:

$$f_c(t) = f(t) + c(t-1)$$

Then, since $\mathbb{E}_Q\left[\frac{P(x)}{Q(x)} - 1\right] = \sum_x (P(x) - Q(x)) = 0$, the corresponding f-divergence remains unchanged:

$$D_{f_c}(P\|Q) = \mathbb{E}_Q\left[f\left(\frac{P(x)}{Q(x)}\right) + c\left(\frac{P(x)}{Q(x)} - 1\right)\right] = D_f(P\|Q)$$

The linear term integrates to zero, so adding such a correction term does not affect the value of the divergence.

This leads to the observation that:

Any convex function f, modified by a linear correction term, still defines the same f-divergence.

3. **Intuition from Taylor Expansion:** In the Taylor series expansion of f, modifying constant or linear terms (first-order approximation) does not affect the divergence. That is, if we change higher-order terms while preserving convexity and keeping $f(1) = 0$, the f-divergence remains invariant:

$$f_c(t) = f(t) + c(t-1) \Rightarrow D_{f_c}(P\|Q) = D_f(P\|Q)$$

Hence, the divergence depends on the curvature (convexity) of the function, not on its linear or constant shifts.

Conclusion:

The f-divergence framework encompasses a wide class of divergence measures. While different convex functions f may define different divergence measures (e.g., KL divergence, total variation, Hellinger, etc.), linear shifts in f do not alter the divergence value. Therefore, when analyzing or constructing divergences, such linear modifications are inconsequential to the informational content of the measure.

(20250211#95)

Obtain expression for family of power functions and its related f-divergence:

We aim to define a family of f-divergences by selecting the power function t^α as a candidate for the convex function f used in the definition:

$$D_f(P\|Q) = \mathbb{E}_Q\left[f\left(\frac{P(x)}{Q(x)}\right)\right]$$

We require that $f(1) = 0$ for consistency with divergence properties.

Checking Convexity:

Let us examine the convexity of the function t^α. Compute its second derivative:

$$\frac{d^2}{dt^2}t^\alpha = \alpha(\alpha-1)t^{\alpha-2}$$

This expression is positive (indicating convexity) when $\alpha(\alpha-1) > 0$, i.e., for $\alpha < 0$ or $\alpha > 1$.

However, this may not be true globally for all α, so we consider instead a scaled version:

$$f(t) = \frac{t^{\alpha-1}-1}{\alpha(\alpha-1)}$$

This function satisfies $f(1) = 0$ and remains convex for suitable values of α.

Degenerate Cases:

- **Case $\alpha \to 0$:** Use L'Hôpital's Rule to compute the limit:

  $$\lim_{\alpha\to 0}\frac{t^\alpha - 1}{\alpha} = \ln t$$

  Therefore,

  $$\lim_{\alpha\to 0}\frac{t^\alpha - 1 - \alpha(t-1)}{\alpha(\alpha-1)} = -\ln t + (t-1)$$

  This expression doesn't change the f-divergence because linear terms do not affect it.

- **Case $\alpha \to 1$:** Apply a correction to handle the degeneracy. We use:

  $$f(t) = \frac{t^\alpha - 1 - \alpha(t-1)}{\alpha(\alpha-1)}$$

  and take the limit as $\alpha \to 1$:

  $$\lim_{\alpha \to 1} f(t) = t \ln t - t + 1$$

  which is a well-known convex function used in defining the Kullback-Leibler (KL) divergence.

Alpha-Logarithm: A generalization of the logarithm function is:

$$\ln_\alpha t = \frac{t^{1-\alpha} - 1}{1-\alpha}$$

This function satisfies:

$$\lim_{\alpha \to 1} \ln_\alpha t = \ln t$$

To make $\ln_\alpha(t)$ concave (like the natural logarithm), we flip the sign:

$$-\ln_\alpha(t) = \frac{1 - t^{1-\alpha}}{1-\alpha}$$

This leads to a concave function for $\alpha > 0$, preserving the shape properties of $\ln t$.

Conclusion: Power functions and their extensions through careful scaling and limiting behavior (as in the case $\alpha \to 0$ and $\alpha \to 1$) provide a rich family of convex functions for defining f-divergences, including KL-divergence as a special case.

(20250211#96)

Sequential binary hypothesis testing problem formulate:

We consider the classical **binary hypothesis testing problem**, where the goal is to distinguish between two hypotheses:

- H_0: $X_1, X_2, \ldots, X_n$ are i.i.d. from P_0
- H_1: $X_1, X_2, \ldots, X_n$ are i.i.d. from P_1

This is a standard **detection problem** in statistics and information theory. However, unlike fixed-sample-size hypothesis testing, we focus here on:

Sequential Hypothesis Testing:

- Instead of fixing n in advance, we sequentially observe data and decide at each step whether to stop or continue sampling.
- The process continues until we are sufficiently confident to make a decision in favor of either H_0 or H_1.

Policy Definition

A sequential decision policy is a collection:

$$\pi = (\pi_n^1, \pi_n^2)_{n \geq 1}$$

Where for each n:

- $\pi_n^1(X_1, \ldots, X_n) \in \{\text{stop}, \text{continue}\}$ determines whether to stop sampling or continue.
- $\pi_n^2(X_1, \ldots, X_n) \in \{H_0, H_1\}$ determines which hypothesis to declare, once stopped.

We define the stopping time:

$$T = \inf \left\{ n \geq 1 \mid \pi_n^1(X_1, \ldots, X_n) = \text{stop} \right\}$$

This is the smallest n such that the policy decides to stop. It could potentially be infinite.

Probability of Error

Let $P_{e,i}^\pi$ denote the probability of making an error when the true hypothesis is H_i, under policy π:

$$P_{e,i}^\pi = \mathbb{P}_i \left(\pi_T^2(X_1, \ldots, X_T) = H_{1-i} \right), \quad i = 0, 1$$

- $P_{e,0}^\pi$ is the probability of false alarm (rejecting H_0 when H_0 is true)
- $P_{e,1}^\pi$ is the probability of missed detection (accepting H_0 when H_1 is true)

Admissibility of Policies

We define the notion of ε-**admissibility** to restrict attention to reasonable policies:

$$\pi \text{ is } \varepsilon\text{-admissible if } P_{e,i}^\pi \leq \varepsilon \quad \text{for } i = 0, 1$$

This rules out degenerate policies like always declaring H_0 or H_1 regardless of observations.

Let Π_ε denote the set of all ε-admissible policies.

Sample Complexity

Another important performance metric is the expected number of samples taken by a policy π:

$$\mathbb{E}_i^\pi[T], \quad i = 0, 1$$

That is, the expected value of the stopping time T under hypothesis H_i.

Tradeoff: There is a fundamental tradeoff between the number of samples drawn and the error probabilities:

- Smaller T (fewer samples) generally increases the risk of error.
- Larger T improves accuracy but at the cost of efficiency.

Distributions under Each Hypothesis

The law of the observed data up to time T depends on the true hypothesis and the policy π:

- Under H_0: $\mathbb{P}_0^\pi = \text{Law}(X_1, \ldots, X_T)$ under P_0
- Under H_1: $\mathbb{P}_1^\pi = \text{Law}(X_1, \ldots, X_T)$ under P_1

Note that since the stopping time T depends on the observations and the policy π, it influences the effective distribution of the sample sequence. Hence, the policy π not only governs decisions but also indirectly determines the data distribution observed at stopping time.

(20250211#97)

State the Wald's theorem:

Setup: Consider the binary hypothesis testing problem:

$$H_0 : X_1, X_2, \ldots \overset{\text{i.i.d.}}{\sim} P_0 \quad \text{vs} \quad H_1 : X_1, X_2, \ldots \overset{\text{i.i.d.}}{\sim} P_1$$

In the sequential setting, instead of fixing the sample size, we design a policy that determines when to stop collecting data and make a decision.

Policy Definition: A policy $\pi = (\pi_n^1, \pi_n^2)_{n \geq 1}$ consists of:

- $\pi_n^1(X_1, \ldots, X_n) \in \{\text{stop, continue}\}$: decides whether to stop sampling.
- $\pi_n^2(X_1, \ldots, X_n) \in \{H_0, H_1\}$: final decision rule upon stopping.

Let the stopping time be

$$T = \inf\{n \geq 1 : \pi_n^1(X_1, \ldots, X_n) = \text{stop}\}$$

Error Probabilities:

$$P^{\pi}_{e,i} = \Pr\left\{\pi^2_T(X_1,\dots,X_T) = H_{1-i} \mid H_i\right\}, \quad i = 0,1$$

A policy π is said to be ε-**admissible** if:

$$P^{\pi}_{e,i} \le \varepsilon, \quad \forall i \in \{0,1\}$$

Let Π_ε denote the set of all ε-admissible policies.

Objective: Minimize the expected sample size under P_i, i.e.,

$$\inf_{\pi\in\Pi_\varepsilon} \mathbb{E}^{\pi}_i[T]$$

Wald's Theorem: For $i = 0$ or 1,

$$\lim_{\varepsilon\to 0} \frac{\inf_{\pi\in\Pi_\varepsilon} \mathbb{E}^{\pi}_i[T]}{\log(1/\varepsilon)} = \frac{1}{D(P_1\|P_0)}$$

This implies:

$$\mathbb{E}^{\pi}_i[T] \ge \frac{\log(1/\varepsilon)}{D(P_i\|P_{1-i})} \quad \text{as } \varepsilon \to 0$$

Interpretation: Any ε-admissible policy must take an increasing number of samples as $\varepsilon \to 0$. No admissible policy can avoid this growth in expected sample size.

(20250211#98)

Prove the Wald's theorem:

Setup: Consider a sequential binary hypothesis testing problem:

$$H_0 : X_1, X_2, \dots \sim P_0 \quad \text{vs.} \quad H_1 : X_1, X_2, \dots \sim P_1$$

We define a policy π consisting of:

- A stopping time T
- A decision rule $Y_T = \pi^2_T(X_1,\dots,X_T) \in \{H_0, H_1\}$

Let the full output of the system be denoted by:

$$(X_1,\dots,X_T) \mapsto (X_1,\dots,X_T,Y_T) = (X,Y)$$

Define the joint distributions:

$$Q^\pi_{i,X,Y} = \text{law of } (X_1, \ldots, X_T, Y_T) \text{ under } H_i \text{ and policy } \pi$$

$$P^\pi_{i,X} = \text{marginal law of } (X_1, \ldots, X_T) \text{ under } H_i$$

Then:

$$D(P^\pi_{0,X} \| P^\pi_{1,X}) = \mathbb{E}^\pi_0 \left[\log \frac{P^\pi_{0,X}(X)}{P^\pi_{1,X}(X)} \right]$$

$$= \mathbb{E}^\pi_0 \left[\log \frac{Q^\pi_{0,X,Y}(X,Y)}{Q^\pi_{1,X,Y}(X,Y)} \right] = D(Q^\pi_{0,X,Y} \| Q^\pi_{1,X,Y})$$

Using the chain rule for KL divergence:

$$D(Q^\pi_{0,X,Y} \| Q^\pi_{1,X,Y}) = D(P^\pi_{0,X} \| P^\pi_{1,X}) + \mathbb{E}^\pi_0 \left[D(Q^\pi_{0,Y|X} \| Q^\pi_{1,Y|X}) \right]$$

By the data processing inequality:

$$D(Q^\pi_{0,X,Y} \| Q^\pi_{1,X,Y}) \geq D(Q^\pi_{0,Y} \| Q^\pi_{1,Y})$$

Decision Distribution Table:

	$Y_T = H_0$	$Y_T = H_1$
H_0	$1 - \varepsilon^\pi_{01}$	ε^π_{01}
H_1	ε^π_{10}	$1 - \varepsilon^\pi_{10}$

$$\varepsilon^\pi_{ij} = \mathbb{P}_i \left\{ \pi^2_T(X_1, \ldots, X_T) = H_j \right\}$$

For an ε-admissible policy, we have:

$$\varepsilon^\pi_{01} \leq \varepsilon, \quad \varepsilon^\pi_{10} \leq \varepsilon \Rightarrow 1 - \varepsilon^\pi_{01} \geq 1 - \varepsilon, \quad 1 - \varepsilon^\pi_{10} \geq 1 - \varepsilon$$

Lower bound on the divergence between decision distributions:

$$\begin{aligned} D(Q^\pi_{0,Y} \| Q^\pi_{1,Y}) &= (1 - \varepsilon^\pi_{01}) \log \left(\frac{1 - \varepsilon^\pi_{01}}{\varepsilon^\pi_{10}} \right) + \varepsilon^\pi_{01} \log \left(\frac{\varepsilon^\pi_{01}}{1 - \varepsilon^\pi_{10}} \right) \\ &= (1 - \varepsilon^\pi_{01}) \log \left(\frac{1}{\varepsilon^\pi_{10}} \right) + (1 - \varepsilon^\pi_{01}) \log (1 - \varepsilon^\pi_{01}) \\ &\quad + \varepsilon^\pi_{01} \log(\varepsilon^\pi_{01}) + \varepsilon^\pi_{01} \log \left(\frac{1}{1 - \varepsilon^\pi_{10}} \right) \\ &\geq (1 - \varepsilon) \log(1/\varepsilon) - c_1 - c_2 - c_3 \end{aligned}$$

for constants c_i that vanish as $\varepsilon \to 0$.

Expected log-likelihood ratio under H_0:

$$\mathbb{E}_0^\pi\left[\log\frac{P_{0,X}^\pi(X)}{P_{1,X}^\pi(X)}\right] = \mathbb{E}_0^\pi\left[\sum_{n=1}^{T}\log\frac{P_0(X_n)}{P_1(X_n)}\right] = \mathbb{E}_0^\pi[T]\cdot D(P_0\|P_1)$$

Putting everything together:

$$\mathbb{E}_0^\pi[T]\cdot D(P_0\|P_1) \geq (1-\varepsilon)\log(1/\varepsilon) - c \Rightarrow \frac{\mathbb{E}_0^\pi[T]}{\log(1/\varepsilon)} \geq \frac{1-\varepsilon-c/\log(1/\varepsilon)}{D(P_0\|P_1)}$$

Taking the infimum over all $\pi\in\Pi_\varepsilon$ and the limit $\varepsilon\to 0$, we get the converse:

$$\liminf_{\varepsilon\to 0}\inf_{\pi\in\Pi_\varepsilon}\frac{\mathbb{E}_0^\pi[T]}{\log(1/\varepsilon)} \geq \frac{1}{D(P_0\|P_1)}$$

Achievability (Direct Part)

Use the log-likelihood ratio test (SPRT):

$$\Lambda_n = \sum_{k=1}^{n}\log\frac{P_0(X_k)}{P_1(X_k)}$$

Stopping rule:

- If $\Lambda_n \geq \log(1/\varepsilon)$, decide H_0
- If $\Lambda_n \leq -\log(1/\varepsilon)$, decide H_1
- Else, continue sampling

If H_0 is true:

$$\Lambda_n \text{ grows positively},\quad \mathbb{E}_0^\pi[T] \leq \frac{\log(1/\varepsilon)}{D(P_0\|P_1)}$$

If H_1 is true:

$$\Lambda_n \text{ tends to } -\infty,\quad \mathbb{E}_1^\pi[T] \leq \frac{\log(1/\varepsilon)}{D(P_1\|P_0)}$$

This policy is ε-admissible:

$$\mathbb{P}_0\left(\text{decide } H_1\right) = \mathbb{P}_0\left(\Lambda_T \leq -\log(1/\varepsilon)\right) \leq \varepsilon$$

Conclusion:

$$\lim_{\varepsilon\to 0}\inf_{\pi\in\Pi_\varepsilon}\frac{\mathbb{E}_0^\pi[T]}{\log(1/\varepsilon)} = \frac{1}{D(P_0\|P_1)}$$

The converse shows we can't do better. The achievability shows Wald's SPRT achieves this bound. **Thus, SPRT is asymptotically optimal.**

(20250213#99)

Show ϵ-admissibility of the sequential hypothesis testing policy.

- Sequential test between hypotheses:

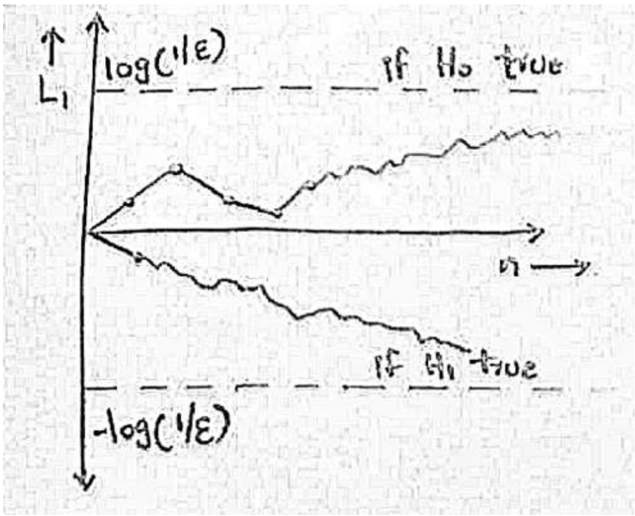
$$\begin{cases} H_0 : X_i \sim P_0 \\ H_1 : X_i \sim P_1 \end{cases}$$

  where observations are i.i.d. under each hypothesis.

Stopping Rule (Sequential Probability Ratio Test):

Let T be the stopping time defined by the log-likelihood ratio process:

$$S_n = \sum_{k=1}^{n} \log \frac{P_1(X_k)}{P_0(X_k)}$$



- If $S_n \geq \log(1/\epsilon)$, then stop and decide H_1
- If $S_n \leq -\log(1/\epsilon)$, then stop and decide H_0
- Otherwise, continue sampling

Expected Stopping Time:

Assuming H_0 is true:

$$\mathbb{E}_0[T] \approx \frac{\log(1/\epsilon)}{D(P_0 \,\|\, P_1)}$$

Type-I Error Analysis:

Type-I error probability is:

$$P_{\epsilon,0} = \mathbb{P}_0(\text{Decide } H_1 \mid H_0 \text{ true}) = \mathbb{P}_0\left(\sum_{i=1}^{T} \log \frac{P_1(X_i)}{P_0(X_i)} \geq \log(1/\epsilon)\right)$$

This can be expressed as:

$$\begin{aligned}
P_{\epsilon,0} &= \mathbb{E}_{P_0}\left[\mathbf{1}\left\{\sum_{i=1}^{T} \log \frac{P_1(X_i)}{P_0(X_i)} \geq \log(1/\epsilon)\right\}\right] \\
&= \mathbb{E}_{P_0}\left[\mathbf{1}\left\{\frac{P_1(X^T)}{P_0(X^T)} \geq \frac{1}{\epsilon}\right\}\right]
\end{aligned}$$

Using **change of measure** from P_0 to P_1:

$$\begin{aligned}
P_{\epsilon,0} &= \mathbb{E}_{P_1}\left[\frac{P_0(X^T)}{P_1(X^T)} \cdot \mathbf{1}\left\{\frac{P_1(X^T)}{P_0(X^T)} \geq \frac{1}{\epsilon}\right\}\right] \\
&\leq \mathbb{E}_{P_1}\left[\epsilon \cdot \mathbf{1}\left\{\frac{P_1(X^T)}{P_0(X^T)} \geq \frac{1}{\epsilon}\right\}\right] \\
&\leq \epsilon
\end{aligned}$$

Why the Inequality Holds:

- When the indicator condition is true:

$$\frac{P_1(X^T)}{P_0(X^T)} \geq \frac{1}{\epsilon} \quad \Longrightarrow \quad \frac{P_0(X^T)}{P_1(X^T)} \leq \epsilon$$

- Thus:

$$\frac{P_0(X^T)}{P_1(X^T)} \cdot \mathbf{1}\left\{\frac{P_1(X^T)}{P_0(X^T)} \geq \frac{1}{\epsilon}\right\} \leq \epsilon$$

- Outside this region, the indicator is zero, so we ignore those samples.

Conclusion:

The sequential test satisfies:

$$P_{\epsilon,0} \leq \epsilon, \quad \mathbb{E}_0[T] \leq \frac{\log(1/\epsilon)}{D(P_0 \,\|\, P_1)}$$

Hence, the policy is ϵ-admissible under the performance criteria for error and average sample size.

(20250213#100)

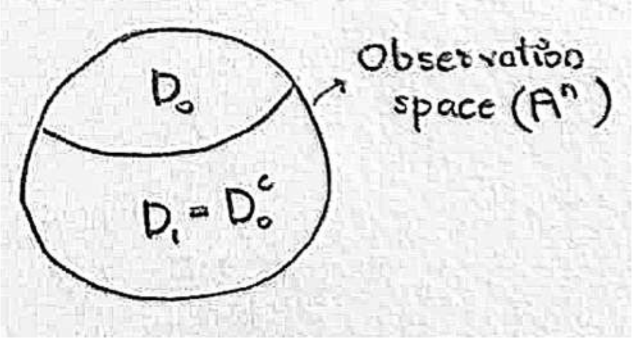
Obtain Chernoff entropy in the context of binary hypothesis testing:

- Hypotheses:

$$H_0 : X_1, X_2, \ldots, X_n \sim \text{i.i.d. } P_0, \quad H_1 : X_1, X_2, \ldots, X_n \sim \text{i.i.d. } P_1$$

- Define a decision region $D_{1,n} \subset \mathcal{A}^n$ such that:

$$x^n \in D_{1,n} \Longrightarrow \text{Decide } H_1$$



Error Probabilities:

- False alarm (Type I error): $\alpha_n = \mathbb{P}_{0,n}(D_{1,n})$
- Missed detection (Type II error): $\beta_n = \mathbb{P}_{1,n}(D^c_{1,n}) = 1 - \mathbb{P}_{1,n}(D_{1,n})$
- Classical bounds:

$$\alpha_n \le \epsilon, \quad \beta_n \approx e^{-nD(P_0 \,\|\, P_1)}$$

  under optimal Neyman–Pearson testing when $\alpha_n \le \epsilon$.

Bayesian Setting with Prior:

- Assume prior probabilities:

$$\pi_0 = \mathbb{P}(H_0), \quad \pi_1 = \mathbb{P}(H_1) = 1 - \pi_0, \quad 0 < \pi_0 < 1$$

- Average probability of error:

$$P_{\text{err}} = \pi_0 \alpha_n + \pi_1 \beta_n$$

- We are interested in the **best exponential decay rate** of the error probability:

$$-\min_{D_{1,n}} \frac{1}{n} \log \epsilon(D_{1,n})$$

  where $\epsilon(D_{1,n}) = P_{\text{err}}$.

Chernoff Theorem:

$$\lim_{n\to\infty} \left(-\min_{D_{1,n}\subset\mathcal{A}^n} \frac{1}{n} \log \epsilon(D_{1,n}) \right) = C(P_0, P_1)$$

where $C(P_0, P_1)$ is the **Chernoff entropy**, which characterizes the optimal exponential error decay.

Chernoff Entropy:

- Defined as:

$$C(P_0, P_1) = \min_{0\le\lambda\le 1} \log \left(\sum_{a\in\mathcal{A}} P_0(a)^{1-\lambda} P_1(a)^{\lambda} \right)$$

- Let $\lambda^* \in [0,1]$ be the minimizer. Then define:

$$P_{\lambda^*}(a) = \frac{P_1(a)^{\lambda^*} P_0(a)^{1-\lambda^*}}{Z_{\lambda^*}}, \quad \text{where } Z_{\lambda^*} = \sum_{a\in\mathcal{A}} P_1(a)^{\lambda^*} P_0(a)^{1-\lambda^*}$$

- This distribution satisfies:

$$D(P_{\lambda^*} \parallel P_0) = D(P_{\lambda^*} \parallel P_1) = C(P_0, P_1)$$

Interpretation:

The Chernoff entropy $C(P_0, P_1)$ gives the tightest achievable exponential decay rate of the Bayes risk $\epsilon(D_{1,n})$ over all decision rules. There is an intrinsic trade-off between false alarm and missed detection: improving one often worsens the other unless the test is finely tuned (as in Neyman–Pearson or Chernoff-optimal rules).

(20250213#101)

What are some properties of Chernoff entropy?

(1) **Example: Binary Hypothesis Testing**

Let the hypothesis distributions be:

$$P_0 \sim \text{Bernoulli}(\theta_0), \quad P_1 \sim \text{Bernoulli}(\theta_1), \quad \mathcal{A} = \{0, 1\}$$

The Chernoff distribution for any $\lambda \in [0, 1]$ is given by:

$$P_\lambda(0) = \frac{(1-\theta_0)^{1-\lambda}(1-\theta_1)^\lambda}{Z_\lambda}, \quad P_\lambda(1) = \frac{\theta_0^{1-\lambda}\theta_1^\lambda}{Z_\lambda}$$

Normalization constant:

$$Z_\lambda = (1-\theta_0)^{1-\lambda}(1-\theta_1)^\lambda + \theta_0^{1-\lambda}\theta_1^\lambda$$

(2) **Parameter Domain**

$$0 \le \lambda \le 1$$

The Chernoff distribution P_λ interpolates between P_0 and P_1 as λ varies:

$$\lambda = 0 \Rightarrow P_\lambda = P_0, \qquad \lambda = 1 \Rightarrow P_\lambda = P_1$$

(3) **Interpretation of P_{λ^*}**

The distribution P_{λ^*}, which achieves the minimum in the Chernoff entropy, can be seen as a statistical "midpoint" between P_0 and P_1, in the sense that:

$$D(P_{\lambda^*} \,\|\, P_0) = D(P_{\lambda^*} \,\|\, P_1)$$

(4) **Symmetry of Chernoff Entropy**

$$C(P_0, P_1) = C(P_1, P_0)$$

(5) **Upper Bound**

$$C(P_0, P_1) \le \min\{D(P_0 \,\|\, P_1),\ D(P_1 \,\|\, P_0)\}$$

This is intuitive since the Chernoff entropy captures the best exponent in the tradeoff between type-I and type-II errors, while KL divergences correspond to the exponent achievable under asymmetric error constraints (Neyman–Pearson).

(20250213#102)

How to prove that the balanced α_n^* and β_n^* in fact corrresponds to the minimum error probability of $\pi_0\alpha_n^* + \pi_1\beta_n^*$:

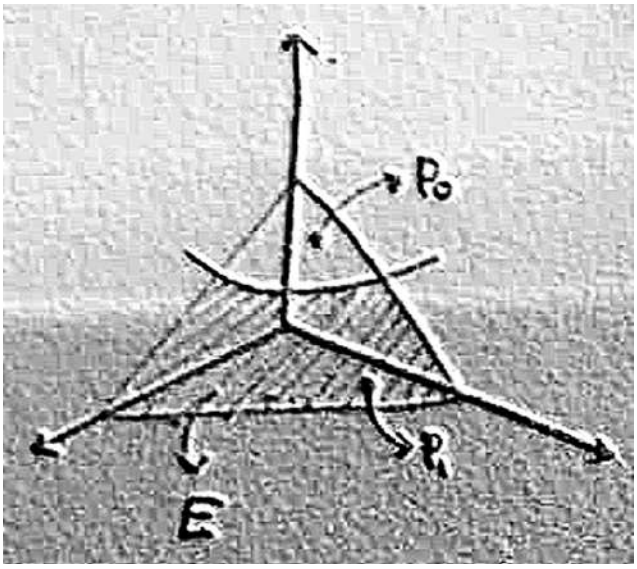
Proof:

Let the total probability of error be:

$$\epsilon(D_{1,n}) = \pi_0 \alpha_n + \pi_1 \beta_n$$

where:

- $\alpha_n = \mathbb{P}_{0,n}(D_{1,n})$: false alarm probability,
- $\beta_n = \mathbb{P}_{1,n}(D_{0,n}) = 1 - \mathbb{P}_{1,n}(D_{1,n})$: missed detection probability.



Our goal is to analyze the exponential decay rate of $\epsilon(D_{1,n})$. The rate is externally determined by the **minimum** of the two terms $\pi_0 \alpha_n$ and $\pi_1 \beta_n$.

Idea: Balance the rate of decay of α_n and β_n

Use the **log-likelihood ratio test** with threshold 0:

$$D_{1,n}^* = \left\{ x^n \in \mathcal{A}^n : \log\left(\frac{P_{1,n}(x^n)}{P_{0,n}(x^n)}\right) \geq 0 \right\}$$

Let α_n^*, β_n^* be the corresponding error probabilities under this threshold test.

Claim: If $\alpha_n < \alpha_n^*$, then $\beta_n > \beta_n^*$. That is, lowering the false alarm rate worsens the missed detection rate, and vice versa.

Proof of Claim:

Suppose we use another test with decision region $D_{1,n}$. Consider the following inequality:

$$\forall x^n \in \mathcal{A}^n, \quad \left(1_{D_{1,n}}(x^n) - 1_{D_{1,n}^*}(x^n)\right)(P_{1,n}(x^n) - P_{0,n}(x^n)) \geq 0$$

This holds due to the Neyman–Pearson lemma: the likelihood ratio test maximizes the detection probability for a given false alarm constraint.

Now summing over all $x^n \in \mathcal{A}^n$, we get:

$$\sum_{x^n} \left(1_{D_{1,n}}(x^n) - 1_{D^*_{1,n}}(x^n)\right) \left(P_{1,n}(x^n) - P_{0,n}(x^n)\right) \geq 0$$

Expanding the terms:

$$\sum_{x^n} 1_{D_{1,n}}(x^n) P_{1,n}(x^n) - \sum_{x^n} 1_{D^*_{1,n}}(x^n) P_{1,n}(x^n) \\ - \left(\sum_{x^n} 1_{D_{1,n}}(x^n) P_{0,n}(x^n) - \sum_{x^n} 1_{D^*_{1,n}}(x^n) P_{0,n}(x^n)\right) \geq 0$$

This simplifies to:

$$(\beta_n^* - \beta_n) + (\alpha_n - \alpha_n^*) \geq 0 \Rightarrow \beta_n \geq \beta_n^* \quad \text{if } \alpha_n < \alpha_n^*$$

Conclusion:

The likelihood ratio test with threshold 0 achieves a balance in the exponential rates of decay for α_n and β_n. Any attempt to improve one (e.g., reducing α_n) necessarily worsens the other (increases β_n), due to the monotonic tradeoff characterized by the Neyman–Pearson lemma.

(20250213#103)

Prove that

$$C(P_0, P_1) \leq \min\{D(P_0||P_1), D(P_1||P_0)\}$$

:

Lower Bound on Error Exponent:

Let the error probability be:

$$\epsilon(D_{1,n}) = \pi_0 \alpha_n + \pi_1 \beta_n \geq \min\{\pi_0, \pi_1\}(\alpha_n + \beta_n)$$

Now, using the optimal likelihood ratio test, we know:

$$\alpha_n + \beta_n \geq \min\{\alpha_n^*, \beta_n^*\} \Rightarrow \epsilon(D_{1,n}) \geq \min\{\alpha_n^*, \beta_n^*\} \cdot \min\{\pi_0, \pi_1\}$$

Taking logarithms and limits:

$$\liminf_{n\to\infty} \frac{1}{n} \log \epsilon(D_{1,n}) \geq \min \left\{ \liminf_{n\to\infty} \frac{1}{n} \log \alpha_n^*, \liminf_{n\to\infty} \frac{1}{n} \log \beta_n^* \right\} = -C(P_0, P_1)$$

The constants $\min\{\pi_0, \pi_1\}$ do not affect the exponential rate, so they vanish in the exponent.

Claim:

$$\lim_{n\to\infty} \frac{1}{n} \log \alpha_n^* = \lim_{n\to\infty} \frac{1}{n} \log(\alpha_n^*)^2 = -C(P_0, P_1)$$

Upper Bound on Error Exponent:

$$\limsup_{n\to\infty} \min_{D_{1,n}} \frac{1}{n} \log \epsilon(D_{1,n}) \leq \limsup_{n\to\infty} \frac{1}{n} \log \epsilon(D_{1,n}^*)$$

Now observe:

$$\epsilon(D_{1,n}^*) = \pi_0 \alpha_n^* + \pi_1 \beta_n^* \leq 2 \cdot \max\{\alpha_n^*, \beta_n^*\} \Rightarrow \log \epsilon(D_{1,n}^*) \leq \log 2 + \max\{\log \alpha_n^*, \log \beta_n^*\}$$

Dividing by n and taking the limit superior:

$$\limsup_{n\to\infty} \frac{1}{n} \log \epsilon(D_{1,n}) \leq \max \left\{ \limsup_{n\to\infty} \frac{1}{n} \log \alpha_n^*, \limsup_{n\to\infty} \frac{1}{n} \log \beta_n^* \right\} = -C(P_0, P_1)$$

Thus,

$$\lim_{n\to\infty} \frac{1}{n} \log \epsilon(D_{1,n}^*) = -C(P_0, P_1)$$

Type-based Analysis of α_n^*:

Consider the optimal decision region:

$$D_{1,n}^* = \left\{ x^n : \log \left(\frac{P_1^n(x^n)}{P_0^n(x^n)} \right) \geq 0 \right\}$$

This is equivalent to:

$$x^n \in D_{1,n}^* \iff D(\tau(x^n) \| P_0) - D(\tau(x^n) \| P_1) \geq 0$$

where $\tau(x^n)$ is the type (empirical distribution) of the sequence x^n.

Now define:

$$\mathcal{E} := \{\tau \in \mathcal{P}(\mathcal{A}) : D(\tau \| P_0) - D(\tau \| P_1) > 0\}$$

Then:

$$\alpha_n^* = \mathbb{P}_{0,n}(D_{1,n}^*) = \mathbb{P}_{0,n}\left(\bigcup_{\tau \in \mathcal{T}_n \cap \mathcal{E}} A_n(\tau)\right) = \sum_{\tau \in \mathcal{T}_n \cap \mathcal{E}} \mathbb{P}_{0,n}(A_n(\tau))$$

From type class bounds:

$$2^{-nD(\tau \| P_0)} \leq \mathbb{P}_{0,n}(A_n(\tau)) \leq (n+1)^{|\mathcal{A}|} 2^{-nD(\tau \| P_0)}$$

Therefore:

$$\alpha_n^* \leq (n+1)^{|\mathcal{A}|} \max_{\tau \in \mathcal{T}_n \cap \mathcal{E}} 2^{-nD(\tau \| P_0)}$$

Taking logs and limits:

$$\lim_{n \to \infty} \frac{1}{n} \log \alpha_n^* = -\min_{\tau \in \mathcal{E}} D(\tau \| P_0) = -C(P_0, P_1)$$

This follows because as $n \to \infty$, the set $\mathcal{T}_n \cap \mathcal{E}$ becomes dense in $\mathcal{E}$, and the minimum over the discrete types approaches the infimum over all distributions in $\mathcal{E}$.

(20250218#104)

How is Chernoff entropy defined?

The Chernoff entropy between two distributions P_0 and P_1 is defined as:

$$C(P_0, P_1) := \min_{P \in \mathcal{E}} D(P \| P_0),$$

where $D(P\|Q)$ denotes the Kullback–Leibler (KL) divergence, and the set $\mathcal{E}$ is:

$$\mathcal{E} := \{P : P \text{ is a PMF on } \mathcal{A} \text{ and } D(P\|P_0) \geq D(P\|P_1)\}.$$

This set $\mathcal{E}$ captures all distributions that are *closer to* P_1 than P_0 in the KL divergence sense.

Asymptotic Error and Large Deviations:

In the context of hypothesis testing, define the Type-I error probability:

$$\alpha_n := \text{Probability of choosing } H_1 \text{ when } H_0 \text{ is true.}$$

Under large deviations theory and using Sanov's Theorem, one can show that:

$$\lim_{n\to\infty} \frac{1}{n} \log \alpha_n = -C(P_0, P_1).$$

Hence, the exponential decay rate of the probability of error (under optimal decision regions) is governed by the Chernoff entropy.

Rewriting the Set $\mathcal{E}$:

Recall that KL divergence can be expressed as:

$$D(P\|P_0) - D(P\|P_1) = \sum_{a\in\mathcal{A}} P(a) \log\left(\frac{P_1(a)}{P_0(a)}\right).$$

Define the function:

$$g(a) := \log\left(\frac{P_1(a)}{P_0(a)}\right).$$

Then the inequality:

$$D(P\|P_0) \geq D(P\|P_1)$$

is equivalent to:

$$\sum_{a\in\mathcal{A}} P(a) g(a) \geq 0, \quad \text{or} \quad \mathbb{E}_P[g(X)] \geq 0.$$

Thus, the set $\mathcal{E}$ can also be written as:

$$\mathcal{E} = \{P \in \mathcal{P}(\mathcal{A}) : \mathbb{E}_P[g(X)] \geq 0\},$$

where $\mathcal{P}(\mathcal{A})$ is the space of PMFs over the finite alphabet $\mathcal{A}$.

Remarks:

- This condition defines a half-space in the space of probability mass functions: the inequality $\mathbb{E}_P[g(X)] \geq 0$ is linear in P.
- The Chernoff entropy corresponds to finding the distribution in this half-space that is closest to P_0 in KL divergence sense.

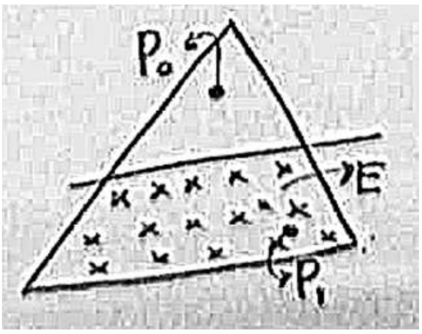
(20250218#105)

Geometrically, where should the minimizer of $D(P\|P_0)$ lie in the probability space?

To understand how Chernoff entropy arises, we begin by considering the framework of hypothesis testing under large deviations.

Suppose we are given two hypotheses:

- H_0 : data is drawn i.i.d. from P_0
- H_1 : data is drawn i.i.d. from P_1



We are interested in characterizing the exponential rate at which the Type I error probability (false positive) decays under optimal decision rules.

Let $\mathcal{T}_n$ denote the set of all **types** (empirical distributions) over sequences of length n. Sanov's Theorem tells us that for any set $\mathcal{A} \subset \mathcal{P}(\mathcal{X})$, the probability of the type class being in $\mathcal{A}$ under i.i.d. P_0 behaves like:

$$\mathbb{P}_{P_0}(\text{type} \in \mathcal{A}) \approx 2^{-n \inf_{P \in \mathcal{A}} D(P\|P_0)}.$$

Let us define the set:

$$\mathcal{E} := \{P \in \mathcal{P}(\mathcal{X}) : D(P\|P_0) \geq D(P\|P_1)\}$$

This is the set of distributions that are closer to P_1 than to P_0, in the KL divergence sense.

Chernoff Entropy is then defined as:

$$C(P_0, P_1) := \min_{P \in \mathcal{E}} D(P\|P_0),$$

which gives the exponent of the decay of the Type I error probability when the alternate hypothesis is P_1, and the decision rule separates sequences that appear closer to P_1 than to P_0.

Geometric Interpretation:

Define the function:

$$g(a) := \log\left(\frac{P_1(a)}{P_0(a)}\right)$$

Then the set $\mathcal{E}$ is equivalently written as:

$$\left\{P : \sum_{a \in \mathcal{A}} P(a) g(a) \geq 0\right\} = \{P : \mathbb{E}_P[g(X)] \geq 0\}$$

This represents a **half-space**, defined by a linear inequality in the space of PMFs. Hence, the separating boundary between the regions closer to P_0 and those closer to P_1 is a hyperplane (i.e., a straight line in 2D), because the constraint is linear in P.

Therefore, the minimizer of $D(P\|P_0)$ within the set $\mathcal{E}$ must lie on the **boundary** of this region—i.e., on the hyperplane defined by $\mathbb{E}_P[g(X)] = 0$.

(20250218#106)

Why is $D(\cdot\|P_0)$ continuous?

Let $P, P_0 \in \mathcal{P}(\mathcal{X})$, where $\mathcal{P}(\mathcal{X})$ denotes the space of all probability mass functions over a finite alphabet $\mathcal{X}$. The Kullback-Leibler (KL) divergence is defined as:

$$D(P\|P_0) := \sum_{x \in \mathcal{X}} P(x) \log \frac{P(x)}{P_0(x)}.$$

We are interested in the continuity of this function with respect to P.

Assumption: Suppose both P_0 and P_1 have **full support** over $\mathcal{X}$, i.e., $P_0(x) > 0$ and $P_1(x) > 0$ for all $x \in \mathcal{X}$.

Claim: Under the full support assumption, $D(P\|P_0)$ is a continuous function of P over $\mathcal{P}(\mathcal{X})$.

Justification:

Consider a sequence $P^{(n)} \to P$ in total variation (or pointwise convergence), i.e., $\forall x \in \mathcal{X}, P^{(n)}(x) \to P(x)$. Because $P_0(x) > 0$ for all x, the function:

$$f_n(x) := P^{(n)}(x) \log \frac{P^{(n)}(x)}{P_0(x)}$$

is well-defined and continuous in $P^{(n)}$ for each fixed x. Moreover, since $\mathcal{X}$ is finite, the sum of finitely many continuous functions remains continuous, implying:

$$D(P^{(n)}\|P_0) \to D(P\|P_0).$$

Hence, $D(P\|P_0)$ is continuous in P as long as P_0 has full support.

Application: Convergence of Error Exponents

In binary hypothesis testing between P_0 and P_1, the Type I error probability α_n and Type II error probability β_n often decay exponentially in the blocklength n. That is,

$$\lim_{n\to\infty} -\frac{1}{n}\log\alpha_n \quad \text{and} \quad \lim_{n\to\infty} -\frac{1}{n}\log\beta_n$$

exist under certain optimal decision rules, and can be expressed using KL divergences between the empirical distribution (type) and the hypotheses.

Earlier, we defined a Chernoff-type bound using:

$$\lim_{n\to\infty} -\frac{1}{n}\log\alpha_n = \min_{P\in\mathcal{E}} D(P\|P_0),$$

where $\mathcal{E}$ is a set of types closer to P_1 than to P_0.

A similar expression exists for β_n, namely:

$$\lim_{n\to\infty} -\frac{1}{n}\log\beta_n = \min_{P\in\mathcal{F}} D(P\|P_1),$$

where $\mathcal{F} = \{P : D(P\|P_1) \geq D(P\|P_0)\}$.

In both cases, the convergence of types $P^{(n)} \to P^* \in \mathcal{E}$ or $\mathcal{F}$ implies convergence of divergences $D(P^{(n)}\|P_0) \to D(P^*\|P_0)$, due to the continuity of KL divergence under the full-support assumption. This guarantees that the error exponents predicted by large deviation theory are accurately realized in the asymptotic regime.

(20250218#107)

Why is it important that we can find a sequence of types $P^{(n)} \in \mathcal{T}_n$ such that $P^{(n)} \to P^*$

as $n \to \infty$? And how does this relate to the convergence of KL divergence $D(P^{(n)} \| P_0) \to D(P^* \| P_0)$?

In the context of hypothesis testing and large deviations, especially when using Sanov's Theorem, we are often interested in approximating a distribution P^* (that minimizes divergence subject to some constraints) using empirical distributions known as *types*. Types form a finite set:

$$\mathcal{T}_n := \left\{ P^{(n)} : P^{(n)}(x) = \frac{N(x; x^n)}{n}, \text{ for some } x^n \in \mathcal{X}^n \right\},$$

where $N(x; x^n)$ is the number of times symbol x appears in the sequence x^n.

Key Observation: Since the space of types $\mathcal{T}_n$ is dense in the space of probability mass functions on a finite alphabet $\mathcal{X}$, we can always find a sequence $P^{(n)} \in \mathcal{T}_n$ such that:

$$P^{(n)} \xrightarrow[n \to \infty]{} P^*.$$

KL Divergence Continuity: Now, consider the KL divergence $D(P \| P_0)$, defined as:

$$D(P \| P_0) = \sum_{x \in \mathcal{X}} P(x) \log \frac{P(x)}{P_0(x)}.$$

If we assume $P_0(x) > 0$ for all $x \in \mathcal{X}$, then this function is continuous with respect to P. Hence, under the convergence $P^{(n)} \to P^*$, we get:

$$D(P^{(n)} \| P_0) \xrightarrow[n \to \infty]{} D(P^* \| P_0).$$

Implication: This result is important in proving the convergence of error exponents in hypothesis testing. For example, if P^* is the distribution in the set:

$$\mathcal{E} := \{ P : D(P \| P_0) \geq D(P \| P_1) \},$$

that minimizes $D(P \| P_0)$, then we can construct a sequence of types $P^{(n)} \in \mathcal{T}_n \cap \mathcal{E}$ such that:

$$\lim_{n \to \infty} D(P^{(n)} \| P_0) = D(P^* \| P_0),$$

thus ensuring that the empirical approximations yield correct asymptotic exponents in testing.

(20250218#108)

Solve the Chernoff entropy optimization problem and state few of its properties:

We are interested in solving the following optimization problem:

$$\min_{P \in \mathcal{E}} D(P \| P_0),$$

where the constraint set is given by:

$$\mathcal{E} = \left\{ P : \sum_a P(a) g(a) \geq 0, \quad \sum_a P(a) = 1 \right\},$$

and the function $g(a) = \log \frac{P_1(a)}{P_0(a)}$, so that

$$\sum_a P(a) g(a) = \mathbb{E}_P \left[\log \frac{P_1(X)}{P_0(X)} \right] = D(P \| P_0) - D(P \| P_1).$$

This corresponds to finding the distribution P that lies in the region closer to P_1 than P_0 in KL sense, and that minimizes divergence from P_0.

Lagrangian Formulation

We use Lagrangian multipliers to incorporate the inequality and equality constraints. Define the Lagrangian:

$$L(P, \lambda, \mu) = D(P \| P_0) - \lambda \sum_a P(a) g(a) + \mu \left(1 - \sum_a P(a) \right),$$

where $\lambda \geq 0$, $\mu \in \mathbb{R}$. This includes:

- The first term $D(P \| P_0) = \sum_a P(a) \log \frac{P(a)}{P_0(a)}$ is convex in P.
- The second term $-\lambda \sum_a P(a) g(a)$ is linear in P.
- The third term $\mu(1 - \sum_a P(a))$ enforces normalization and is also linear.

Thus, the total objective is a convex function in P, which makes the optimization well-posed.

Finding the Minimizer $P_{\lambda,\mu}$

To find the minimizing P, we differentiate the Lagrangian with respect to $P(a)$ and set it to zero:

$$\frac{\partial L}{\partial P(a)} = \log \frac{P(a)}{P_0(a)} + 1 - \lambda g(a) - \mu = 0.$$

Solving this gives:

$$\log \frac{P(a)}{P_0(a)} = \lambda g(a) + \mu - 1 \Rightarrow \frac{P(a)}{P_0(a)} = \exp(\lambda g(a) + \mu - 1),$$

$$\Rightarrow P(a) = P_0(a) \cdot \exp(\lambda g(a) + \mu - 1).$$

Since $g(a) = \log \frac{P_1(a)}{P_0(a)}$, we have:

$$P(a) = P_0(a) \cdot \left(\frac{P_1(a)}{P_0(a)}\right)^{\lambda} \cdot e^{\mu-1} = e^{\mu-1} \cdot P_1(a)^{\lambda} \cdot P_0(a)^{1-\lambda}.$$

Hence, we obtain the Chernoff distribution:

$$P_\lambda(a) \propto P_1(a)^\lambda P_0(a)^{1-\lambda}.$$

The scalar $e^{\mu-1}$ ensures that $\sum_a P(a) = 1$.

Constraint Satisfaction and Optimality

If we can find $\lambda^* \geq 0$, $\mu^* \in \mathbb{R}$ such that:

$$\sum_a P_{\lambda^*,\mu^*}(a) = 1,$$
$$\sum_a P_{\lambda^*,\mu^*}(a) g(a) = 0,$$

then $P_{\lambda^*,\mu^*} \in \mathcal{E}$, and satisfies the Karush-Kuhn-Tucker (KKT) conditions.

This means P_{λ^*} is the optimal solution:

$$\min_{P \in \mathcal{E}} D(P \| P_0).$$

Furthermore, the value:

$$D(P_{\lambda^*} \| P_0) - D(P_{\lambda^*} \| P_1) = \sum_a P_{\lambda^*}(a) g(a) = 0,$$

i.e., the Chernoff distribution is equidistant (in KL sense) from both P_0 and P_1.

The Chernoff-type distribution has the form:

$$P_{\lambda,\mu}(a) = \frac{P_1(a)^\lambda P_0(a)^{1-\lambda}}{Z(\lambda)},$$

where the normalizing constant $Z(\lambda)$ ensures $\sum_a P_{\lambda,\mu}(a) = 1$, and is given by:

$$Z(\lambda) = \sum_{a \in \mathcal{A}} P_1(a)^\lambda P_0(a)^{1-\lambda}.$$

The corresponding parameter μ satisfies:

$$2^{\mu - \log e} = \frac{1}{Z(\lambda)}.$$

Four Properties of the Chernoff Distribution $P_{\lambda,\mu}$:

1. The distribution $P_{\lambda,\mu}$ exists and is non-negative for all a.
2. There exists $\lambda^* \geq 0$, $\mu^* \in \mathbb{R}$ such that:
$$\sum_a P_{\lambda^*,\mu^*}(a) g(a) = 0, \quad \sum_a P_{\lambda^*,\mu^*}(a) = 1.$$
3. The parameter μ^* is chosen to normalize the distribution:
$$\sum_a P_1(a)^\lambda P_0(a)^{1-\lambda} = 1 \quad \Rightarrow \quad 2^{\mu^* - \log e} = \left(\sum_a P_1(a)^\lambda P_0(a)^{1-\lambda} \right)^{-1}.$$
4. The parameter λ parametrizes the path between P_0 and P_1 on the exponential family curve.

Balanced Point of KL Divergence:

We define the function:

$$f(\lambda) = D(P_\lambda \| P_0) - D(P_\lambda \| P_1) = \sum_a P_\lambda(a) g(a).$$

This function has the properties:

- $f(0) = -D(P_0 \| P_1)$,
- $f(1) = D(P_1 \| P_0)$,
- $f(\lambda^*) = 0$, when the distribution is equidistant from both P_0 and P_1.

We also note that:

$$\frac{d}{d\lambda} f(\lambda) = \mathrm{Var}_{P_\lambda} \left[\log \frac{P_1(X)}{P_0(X)} \right] \geq 0,$$

so $f(\lambda)$ is not only continuous but also non-decreasing.

Hence, by the intermediate value theorem, there exists a $\lambda^* \in [0, 1]$ such that $f(\lambda^*) = 0$, and:

$$D(P_{\lambda^*} \| P_0) = D(P_{\lambda^*} \| P_1).$$

This critical point P_{λ^*} is used to define the **Chernoff information**:

$$C(P_0, P_1) = \min_{0 \le \lambda \le 1} D(P_\lambda \| P_0) = D(P_{\lambda^*} \| P_0).$$

(20250218#109)

What is the physical significance of Chernoff entropy?

Consider a binary hypothesis testing problem between the null hypothesis $H_0 \sim P_0$ and the alternative hypothesis $H_1 \sim P_1$, where the decision is made based on n i.i.d. samples. Define:

- α_n: probability of type-I error (reject H_0 when it is true),
- β_n: probability of type-II error (accept H_0 when H_1 is true),
- $\pi_0, \pi_1 \in (0, 1)$: prior probabilities of H_0 and H_1, respectively.

Bayes Risk: The total probability of error under the Bayesian setting is:

$$\pi_0 \alpha_n + \pi_1 \beta_n.$$

Depending on how we control α_n and β_n, this risk behaves differently in the asymptotic regime. The following bounds hold under optimal hypothesis testing strategies.

Case 1: Fix $\alpha_n \le \alpha$, minimize β_n

Then, as $n \to \infty$, the best possible error exponent is given by the **Kullback-Leibler divergence**:

$$\beta_n \approx e^{-nD(P_0 \| P_1)}.$$

Case 2: Fix $\beta_n \le \beta$, minimize α_n

Similarly, we have:

$$\alpha_n \approx e^{-nD(P_1 \| P_0)}.$$

Chernoff Bound:

The **Chernoff information** captures the best exponent achievable when both α_n and β_n decay exponentially, and the total risk is minimized:

$$\pi_0 \alpha_n + \pi_1 \beta_n \approx e^{-nC(P_0, P_1)},$$

where

$$C(P_0, P_1) = \min_{0 \le \lambda \le 1} D(P_\lambda \| P_0) = D(P_{\lambda^*} \| P_0) = D(P_{\lambda^*} \| P_1),$$

and $P_\lambda \propto P_0^{1-\lambda} P_1^\lambda$ is the exponential family mixture between P_0 and P_1.

Interpretation:

- The term $\pi_0 \alpha_n + \pi_1 \beta_n$ is the expected total error, which decays exponentially with n under the optimal decision rule.
- The exponent $C(P_0, P_1)$ represents the tightest achievable exponent under symmetric treatment of both error types.
- Depending on whether the goal is to bound α_n or β_n, the divergence $D(P_0 \| P_1)$ or $D(P_1 \| P_0)$ governs the rate of exponential decay.
- The Chernoff information gives a universal lower bound on the exponential rate of the Bayes error probability:
$$\pi_0 \alpha_n + \pi_1 \beta_n \gtrapprox e^{-nC(P_0, P_1)}.$$

(20250218#110)

Define mutual information in the context of binary hypothesis testing:

We revisit the definition and interpretation of **mutual information** by formulating it as a **binary hypothesis testing problem**, known as *independence testing.*

Setup:

Let $(X, Y) \sim P_{XY}$ be a pair of random variables jointly distributed over some finite alphabet $\mathcal{X} \times \mathcal{Y}$. Let P_X and P_Y denote the corresponding marginal distributions.

We consider the following hypothesis test:

- **Null Hypothesis ($\mathbf{H}_0$):** $X \perp Y$, i.e., the joint distribution is the product of marginals:
$$\widetilde{P}_{XY} = P_X \cdot P_Y.$$
- **Alternate Hypothesis ($\mathbf{H}_1$):** X and Y are dependent, specifically:
$$P_{XY} \neq P_X \cdot P_Y.$$
We assume the alternative distribution is known and is equal to the true joint distribution P_{XY}.

Error Probabilities in Hypothesis Testing:

Given n i.i.d. samples (X^n, Y^n), we define:

- α_n: probability of type-I error (rejecting H_0 when H_0 is true),
- β_n: probability of type-II error (accepting H_0 when H_1 is true).

Under standard large deviations analysis (e.g., Sanov's theorem), if we fix $\beta_n \leq \beta$, then the best achievable exponent for the type-I error probability is given by the Kullback-Leibler divergence between P_{XY} and $P_X \cdot P_Y$:

$$\alpha_n \approx e^{-nD(P_{XY}\|P_XP_Y)}.$$

Definition: Mutual Information

This leads to the natural definition of **mutual information** as:

$$I(X;Y) \triangleq D(P_{XY}\|P_XP_Y),$$

which quantifies the amount of statistical dependence between X and Y. It also has the operational meaning of being the optimal error exponent in the independence testing problem under asymmetric constraints.

Remark: Lautum Information

If we reverse the roles of H_0 and H_1, i.e., suppose the true distribution is the independent one $P_X \cdot P_Y$, and the alternate hypothesis is the dependent distribution P_{XY}, then the corresponding exponent would be:

$$D(P_XP_Y\|P_{XY}),$$

which is known as the **Lautum information**. This is analogous to flipping the roles of type-I and type-II errors in the hypothesis testing setup.

(20250218#111)

Give some properties of mutual information:

- **1. Non-negativity of Mutual Information:**

  $$I(X;Y) \geq 0$$

  Mutual information is always non-negative.

- **2. Zero Mutual Information:**

  $$I(X;Y) = 0 \iff X \perp Y$$

  Mutual information between two random variables is zero if and only if they are independent (i.e., $X \perp Y$).

- **3. Writing Relative Entropy as Expectation of Log Likelihood Ratios:**

  $$I(X;Y) = \mathbb{E}_{P_{XY}} \left[\log \left(\frac{P_{XY}(x,y)}{P_X(x)P_Y(y)} \right) \right]$$

  This expresses mutual information as the expected value of the logarithm of the likelihood ratio between the joint distribution P_{XY} and the product of the marginal distributions P_X and P_Y.

  $$= \mathbb{E}_{P_{XY}} \left[\log \left(\frac{P_{X|Y}(x|y)}{P_X(x)} \right) \right]$$

  This representation relates mutual information to the conditional distribution $P_{X|Y}$.

  $$= \mathbb{E}_{P_Y} \left[\mathbb{E}_{P_{X|Y}} \left[\log \left(\frac{P_{X|Y}(x|y)}{P_X(x)} \right) \right] \right]$$

  By conditioning on Y, we express mutual information as an expectation over P_Y and a nested expectation over $P_{X|Y}$.

  $$= \mathbb{E}_{P_X} \left[\mathbb{E}_{P_{Y|X}} \left[\log \left(\frac{P_{Y|X}(y|x)}{P_Y(y)} \right) \right] \right]$$

  Similarly, we can express mutual information in terms of the conditional distribution $P_{Y|X}$.

- **4. Mutual Information and Entropies:**

  $$I(X;Y) = H(X) + H(Y) - H(X,Y)$$

  This equation relates mutual information to the entropies of X, Y, and their joint distribution.

  $$= H(X) - (H(X,Y) - H(Y)) = H(X) - H(X|Y)$$

  This expresses mutual information as the difference between the entropy of X and the conditional entropy $H(X|Y)$, which captures the reduction in uncertainty of X given Y.

  $$\text{By symmetry, this is also } H(Y) - H(Y|X)$$

  By symmetry, mutual information can also be written as $H(Y) - H(Y|X)$.

- **5. Symmetry of Mutual Information:**

  $$I(X;Y) = I(Y;X) = I(X \wedge Y)$$

  Mutual information is symmetric, meaning that the amount of information X provides about Y is equal to the amount of information Y provides about X.

- **6. Source Coding Interpretation:** Let $A = \{1, 2, \ldots, |A|\}$ be a set of sources. Each source corresponds to a different conditional distribution $P_{Y|X}(\cdot|a)$, where $a \in A$. The random variable X indicates which source is chosen, and Y represents the outcome to be compressed.

  The probability simplex over A defines a set of distributions from which we choose a distribution Q. The length function L_Q is defined as:

  $$L_Q = \lceil \log \frac{1}{Q(c_a)} \rceil$$

  For large code lengths, we approximate L_Q as:

  $$L_Q = \log \frac{1}{Q(c_a)}$$

  The redundancy of the code is given by:

  $$\text{Redundancy} = \mathbb{E}_{P_{Y|X}(\cdot|a)}[L_Q(Y)] - H(P_{Y|X}(\cdot|a))$$

  This measures the difference between the expected code length and the entropy of the source.

  The average redundancy over all sources is:

  $$\text{Average Redundancy} = \sum_a P_X(a) \cdot D(P_{Y|X}(\cdot|a) \| Q)$$

  where $D(P\|Q)$ is the Kullback-Leibler divergence between distributions P and Q.

  $$= D(P_{Y|X} \| Q / P_X)$$

  The average redundancy is the divergence between the conditional distribution $P_{Y|X}$ and the assumed distribution Q, weighted by the distribution of X.

- **Finding Optimal Q:** To minimize redundancy, we seek the distribution Q that minimizes the Kullback-Leibler divergence:

  $$\min_Q D(P_Y \| Q)$$

  This corresponds to finding the centroid of all the distributions $P(1), P(2), \ldots, P(n)$.

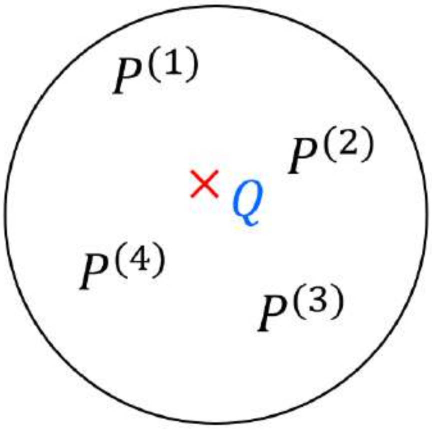
1 20250222

(20250222#112)

In the context of information radius and information centre, given priors Π_is how to pick the Q such that average redundancy is minimized?

Information radius is also called Jensen-Shannon divergence.

Continuing from the previous lecture, here we're treating $P^{(i)} = P_{Y|X}(Y|X = x_i)$. Depending on the source value $(X = x_i)$, we get a probability distribution $P^{(i)}$. All these distributions are defined for symbols taken from the alphabet set B.



Use Q to be representative for all of $P^(1), \ldots, P^(4)$.

If we were to encode by assuming a distribution Q, the expected compressed length would be

$$\mathbb{E}_{P^{(i)}}\left[\log \frac{1}{Q(Y)}\right]$$

Redundancy:

$$\mathbb{E}_{P^{(i)}}\left[\log \frac{1}{Q(Y)}\right] - \mathbb{E}_{P^{(i)}}\left[\log \frac{1}{P^{(i)}(Y)}\right] = D(P^{(i)}||Q)$$

This mismatch is a result of using an assumed distribution $Q(Y)$ instead of the true distribution $P^{(i)}(Y)$.

Average redundancy:

$$\sum_{i}^{I} \Pi_i D(P^{(i)}||Q)$$

where we've tacitly lifted from alphabet to n realization here. Source occurs with probability Π_i → Bayesian world assumption.

Given the priors Π_is, how do we pick the Q?

This is done by minimizing the average redundancy. Minimizing it would mean that the assumed distribution is as close to the true distributions as possible under the given priors.

$$\min_Q \sum_i \Pi_i D(P^{(i)}||Q) = \min_Q D(P_{Y|X} \mid\mid Q \mid \Pi_x), \quad \text{as} \quad P_{Y|X=x_i} = P^{(i)}$$
$$= \min_Q \mathbb{E}\left[\log \frac{P_{Y|X}(Y|X)}{Q(Y)}\right]$$

(20250222#113)

Show that the minimum average redundancy is $I(X;Y)$ and the minimizer has the form:

$$Q^*(y) = P_Y(y) = \sum_{a\in\mathcal{A}} \Pi_X(a)P^{(a)}(y) = \sum_{a\in\mathcal{A}} \Pi_a P^{(a)}(y)$$

Let $\mathcal{A}$ be the alphabet of the source X, and let $P^{(a)} = P_{Y|X=a}$ be the conditional distribution of Y given $X = a \in \mathcal{A}$.

Assume $X \sim \Pi_X$, i.e., the source emits symbol $a \in \mathcal{A}$ with probability $\Pi_X(a)$. Then, the optimal choice of Q that minimizes the average redundancy is:

$$Q^*(y) = P_Y(y) = \sum_{a\in\mathcal{A}} \Pi_X(a)P^{(a)}(y) = \sum_{a\in\mathcal{A}} \Pi_a P^{(a)}(y)$$

This is the marginal distribution of Y when $(X,Y) \sim \Pi_X \cdot P_{Y|X}$.

Minimum Average Redundancy

The expected redundancy when using a mismatched distribution Q instead of the true $P^{(a)}$ is:

$$\sum_{a\in\mathcal{A}} \Pi_a D(P^{(a)}\|Q)$$

Minimizing this over all Q gives:

$$\min_Q \sum_{a\in\mathcal{A}} \Pi_a D(P^{(a)}\|Q) = D(P_{Y|X}\|Q \mid \Pi_X) = \mathbb{E}_{\Pi_X P_{Y|X}}\left[\log \frac{P_{Y|X}(Y|X)}{Q(Y)}\right]$$

Setting $Q = P_Y$ (the marginal of Y), the minimum redundancy becomes:

$$\min_Q \sum_{a\in\mathcal{A}} \Pi_a D(P^{(a)}\|Q) = I(X;Y)$$

Thus, the minimum average redundancy is equal to the mutual information between X and Y.

In the case of n-length sequences, this generalizes to:

$$\frac{1}{n} I(X; Y_1, Y_2, \ldots, Y_n)$$

Proof Sketch

We begin with the identity for conditional relative entropy:

$$D(P_{Y|X} \| Q \mid \Pi_X) - D(P_{Y|X} \| P_Y \mid \Pi_X)$$

By chain rule of KL divergence, this equals:

$$\mathbb{E}_{\Pi_X P_{Y|X}} \left[\log \frac{P_{Y|X}(Y|X)}{Q(Y)} - \log \frac{P_{Y|X}(Y|X)}{P_Y(Y)} \right] = \mathbb{E}_{\Pi_X P_{Y|X}} \left[\log \frac{P_Y(Y)}{Q(Y)} \right]$$

Since $(X, Y) \sim \Pi_X P_{Y|X}$, the marginal of Y is P_Y. Therefore:

$$\mathbb{E}_{P_Y} \left[\log \frac{P_Y(Y)}{Q(Y)} \right] = D(P_Y \| Q) \geq 0$$

Equality holds when $Q = P_Y$, which completes the proof.

Interpretation

To minimize redundancy, we should select the encoding distribution Q to match the marginal distribution of Y:

$$Q^*(y) = P_Y(y) = \sum_{a \in \mathcal{A}} \Pi_a P^{(a)}(y)$$

This is a weighted average of the conditional distributions $P^{(a)}$, where the weights are the source priors Π_a. In other words, the optimal assumed distribution is the average behavior of the true conditional distributions under the source distribution.

(20250222#114)

Remark on universal source coding and mixture distributions:

---

(1) In the context of universal source coding:

In universal source coding, particularly when dealing with n-length strings $X^n = (X_1, X_2, \ldots, X_n)$, we do not assume prior knowledge of the exact source distribution. Instead, we compress using a *universal* distribution, often denoted by Q.

- In the standard universal source coding framework, we typically assume that Q is **i.i.d.**, i.e., $Q(X^n) = \prod_{i=1}^n Q(X_i)$.
- However, in scenarios involving a mixture of source distributions (e.g., the Bayesian case where X has prior Π_X and conditional $P_{Y|X}$), the observed data Y^n is generated according to a *mixture distribution*:

  $$Q(Y) = \sum_{a \in \mathcal{A}} \Pi_X(a) P^{(a)}(Y), \quad \text{where } P^{(a)} = P_{Y|X=a}.$$

  This leads to a mixed distribution over Y, and hence over X^n, that is not i.i.d. in general.

Redundancy Interpretation:

The redundancy incurred in this universal setup is due to using a universal distribution that does not match the true n-string distribution. Specifically, we saw that the redundancy:

$$\min_Q \sum_i \Pi_i D(P^{(i)} \| Q) = I(X;Y),$$

arises because we encode using the mixture $Q = P_Y$, instead of encoding separately for each $P^{(i)}$.

Length of Encoded Sequence:

For a given n-length sequence X^n, suppose the length assigned by the code is:

$$\text{Length}(X^n) = \log\left(\frac{1}{\tilde{Q}(X^n)}\right),$$

where $\tilde{Q}(X^n)$ is the effective universal code probability assignment.

Note:

$$\tilde{Q}(X^n) = 2^{-\text{Length}(X^n)},$$

This $\tilde{Q}(X^n)$ behaves like a *probability assignment*, but it is not necessarily a valid probability distribution, as it may not sum to 1 over all X^n. That is:

$$\sum_{X^n} \tilde{Q}(X^n) \leq 1.$$

Such $\tilde{Q}$ is referred to as a **universal probability assignment**, used in Minimum Description Length (MDL) theory and other universal coding contexts.

Summary:

- In the mixture model, the effective distribution used for coding is not i.i.d., but a mixture of i.i.d. sources.

- The loss in compression performance—quantified by the mutual information $I(X;Y)$—is a consequence of encoding with a universal distribution Q rather than using the true conditional distributions $P^{(i)}$.
- The assigned length satisfies Kraft's inequality due to the prefix-free code, but the corresponding $\tilde{Q}(X^n)$ is not a proper probability distribution.

(20250222#115)

What would be the encoding lengths for the n-length source string scenario where the source distribution is unknown and the source may be drawn from a family of i.i.?

In the context of universal source coding, particularly for sequences of length n, the key idea is to assign a coding length to $x^n = (x_1, x_2, \ldots, x_n)$ without assuming full knowledge of the true source distribution. When the source may be drawn from a family of i.i.d. distributions, the universal code must efficiently represent *all* such possible sources. This requires stepping **outside the i.i.d. assumption** to describe an efficient code for every possible i.i.d. distribution.

Universal Code Length Assignment:

For a universal code, we define the length of a sequence x^n via a function:

$$\text{Length}(x^n) = \log\left(\frac{1}{\tilde{Q}(x^n)}\right),$$

where $\tilde{Q}(x^n)$ is not necessarily a valid probability distribution, but rather a **universal probability assignment**.

This assignment is such that:

$$\tilde{Q}(x^n) = 2^{-\text{Length}(x^n)}.$$

This assignment satisfies Kraft's inequality:

$$\sum_{x^n} 2^{-\text{Length}(x^n)} \leq 1,$$

ensuring the code is prefix-free. However, $\tilde{Q}$ is not necessarily an i.i.d. distribution and may not sum to 1, so it is not a true probability distribution. Nonetheless, it is used for analysis and design of universal source codes.

Why Step Outside the i.i.d. Model?

Suppose we want to efficiently compress sequences generated by some i.i.d. source $P^{(i)}$ drawn from a family $\{P^{(1)}, \dots, P^{(k)}\}$. One option is to take a mixture distribution:

$$Q = \sum_i \Pi_i P^{(i)},$$

and use $Q^{\otimes n}$ to assign code lengths. But this approach leads to **redundancy**:

$$\sum_i \Pi_i D(P^{(i)} \| Q) = I(X; Y),$$

which reflects the mismatch between each $P^{(i)}$ and the mixture Q.

To eliminate this redundancy, we need a mechanism to **universally** assign code lengths that approximate the performance of using the optimal distribution for each $P^{(i)}$, without knowing which one generated the data.

This is only possible if we go beyond i.i.d. representations—i.e., by constructing $\tilde{Q}(x^n)$ that is not constrained to be i.i.d.—because no single i.i.d. distribution can simultaneously match all possible $P^{(i)}$ distributions over sequences.

(20250222#116)

What would the redundancy associated with a representative distribution of a family of distributions be, if we didn't know the priors?

In the Bayesian formulation of universal source coding, we assume that each source distribution $P^{(i)}$ occurs with some known prior probability π_i. These priors allow us to define the average redundancy when we encode using an assumed distribution Q:

$$\text{Average Redundancy} = \sum_i \pi_i D(P^{(i)} \| Q).$$

However, in many situations, we do *not* have access to the prior probabilities π_i. That is, we have no Bayesian justification to weigh the different source models. This leads to the question:

How do we design Q when no prior information about the source model is available?

Worst-Case Redundancy:

In the absence of priors, a natural goal is to minimize the **worst-case** redundancy, i.e., the maximum redundancy over all possible sources. Formally, this is defined as:

$$\max_{\pi_i} \sum_i \pi_i D(P^{(i)} \| Q) = \max_i D(P^{(i)} \| Q),$$

where the maximum is achieved when all the probability mass is concentrated on the worst source. Hence, the problem becomes:

$$\min_Q \max_i D(P^{(i)} \| Q).$$

This is known as the **minimax redundancy** problem: we choose Q to minimize the maximum divergence from all possible source distributions.

Geometric Interpretation:

This optimization has a natural geometric interpretation. Suppose each $P^{(i)}$ is a point in the probability simplex (i.e., the space of all probability distributions over the alphabet $\mathcal{B}$). Then, each divergence $D(P^{(i)} \| Q)$ can be thought of as a squared distance (though not symmetric) between $P^{(i)}$ and Q.

We wish to place the "center" Q such that:

- All the source distributions $P^{(i)}$ lie within a "divergence ball" centered at Q.
- The radius (maximum divergence) of this ball is minimized.

Hence, we are looking for the center Q that forms the **smallest possible divergence ball** that still encloses all $P^{(i)}$.

Formal Statement (Worst-Case Redundancy):

$$\min_Q \max_{\pi_X} D(P_{Y|X} \| Q \mid \pi_X),$$

where $P^{(i)} = P_{Y|X=x_i}$ and π_X ranges over all possible priors on $\mathcal{X}$. That is, we are minimizing the conditional divergence (i.e., the expected redundancy) over the worst possible choice of π_X.

Summary:

- When priors π_i are not available, we move from Bayesian to worst-case analysis.
- The goal is to choose Q to minimize the maximum redundancy:

$$\min_Q \max_i D(P^{(i)} \| Q).$$

- Geometrically, this corresponds to finding the smallest "ball" (in KL-divergence) that contains all $P^{(i)}$.
- This formulation leads to the **information radius** (or Jensen-Shannon divergence) as a fundamental quantity.

(20250222#117)

What happens when we flip the min and max in the min-max redundancy problem?

We have seen that in the absence of priors over the sources, we consider the worst-case redundancy:

$$\min_{Q} \max_{\pi_X} D(P_{Y|X} \| Q \mid \pi_X),$$

where π_X is the distribution over source indices (or source types), and Q is the codebook distribution.

Now consider flipping the order of the min and max operations:

$$\max_{\pi_X} \min_{Q} D(P_{Y|X} \| Q \mid \pi_X).$$

This means that:

- The **adversary** picks a source distribution π_X first,
- Then the **compressor** picks a Q optimally for that chosen π_X,
- The adversary tries to maximize the best performance the compressor can achieve,
- This leads to:

  $$\max_{\pi_X} \min_{Q} D(P_{Y|X} \| Q \mid \pi_X) = \max_{\pi_X} I(X;Y),$$

  since the minimum of the conditional divergence is achieved when $Q = P_Y = \sum_x \pi_X(x) P_{Y|X=x}$, giving the mutual information.

Game-Theoretic Viewpoint:

This optimization can be interpreted as a **two-player zero-sum game**:

- **Player 1 (Maximizer / Adversary):** Chooses π_X, aiming to maximize the redundancy,
- **Player 2 (Minimizer / Compressor):** Chooses Q based on π_X, trying to minimize the redundancy.

Key Insight: The player who chooses second has an advantage because they can react optimally to the opponent's strategy.

- In the **minimax case**:
$$\min_{Q}\max_{\pi_X} D(P_{Y|X}\|Q \mid \pi_X),$$
the compressor chooses Q *after* the adversary chooses π_X. So the compressor has the advantage.

- In the **maximin case**:
$$\max_{\pi_X}\min_{Q} D(P_{Y|X}\|Q \mid \pi_X),$$
the adversary reacts to the compressor's choice, giving the advantage to the adversary.

Comparison:

$$\min_{Q}\max_{\pi_X} D(P_{Y|X}\|Q \mid \pi_X) \quad \text{vs.} \quad \max_{\pi_X}\min_{Q} D(P_{Y|X}\|Q \mid \pi_X)$$

In general:
$$\min\max \leq \max\min,$$
so the minimax redundancy (worst-case) is less than or equal to the maximin redundancy (Bayesian case with unknown prior).

Conditions for Equality:

The min and max operations can be interchanged (i.e., minimax = maximin) when:

- The function is convex in the minimization variable (Q),
- Concave in the maximization variable (π_X),
- The domain of optimization is closed, bounded, and convex,
- These conditions are typically satisfied in our setting since:
$$D(P_{Y|X}\|Q \mid \pi_X) = \sum_{x} \pi_X(x) D(P^{(x)}\|Q)$$
is linear (hence concave) in π_X and convex in Q.

Summary:

- **Minimax:** Encoder assumes no knowledge of nature's choice (π_X). This gives an upper bound on redundancy.
- **Maximin:** Encoder knows nature's move (i.e., knows π_X) and can adapt optimally.
- **Game interpretation:** The adversary tries to maximize redundancy; the compressor tries to minimize it.

- The advantage lies with the player who moves second.

(20250222#118)

Justify that this map is convex:

$$Q \mapsto D(P_{Y|X} \,\|\, Q \mid \pi_X)$$

We are interested in the following optimization:

$$\min_{Q} \max_{\pi_X} D(P_{Y|X} \,\|\, Q \mid \pi_X),$$

where $D(P_{Y|X} \,\|\, Q \mid \pi_X)$ is the conditional Kullback–Leibler divergence:

$$D(P_{Y|X} \,\|\, Q \mid \pi_X) = \sum_x \pi_X(x) D(P_{Y|X=x} \,\|\, Q).$$

Key Claim: The function

$$Q \mapsto D(P_{Y|X} \,\|\, Q \mid \pi_X)$$

is convex in Q.

Justification:

We rely on the following known result from information theory:

- The Kullback–Leibler divergence

  $$(P, Q) \mapsto D(P \,\|\, Q)$$

  is jointly convex in (P, Q).

- This means that for any $0 \leq \lambda \leq 1$, and for any pairs of distributions (P_1, Q_1) and (P_2, Q_2),

  $$D\left(\lambda P_1 + (1-\lambda) P_2 \,\|\, \lambda Q_1 + (1-\lambda) Q_2\right) \leq \lambda D(P_1 \,\|\, Q_1) + (1-\lambda) D(P_2 \,\|\, Q_2).$$

- In our case, we fix the first argument $P = P_{Y|X=x}$ (i.e., the channel), and optimize over Q.

- Since fixing P and varying Q retains the convexity of $D(P \,\|\, Q)$ in Q, we conclude that:

  $$Q \mapsto D(P_{Y|X} \,\|\, Q \mid \pi_X) = \sum_x \pi_X(x) D(P_{Y|X=x} \,\|\, Q)$$

  is a convex combination of convex functions in Q, and hence convex.

Interpretation:

- The optimization problem $\min_Q \max_{\pi_X} D(P_{Y|X} \| Q \mid \pi_X)$ is a **convex minimization** problem in Q, for fixed π_X.
- This convexity property is essential to ensure well-behaved optimization (i.e., existence of global minima, no local minima issues).
- Since $P_{Y|X}$ defines a family of distributions $\{P^{(x)}\}$ over the output alphabet, and π_X is a distribution over inputs x, the expected divergence is convex in Q even though it's aggregated over x.

Summary:

- The divergence function $Q \mapsto D(P_{Y|X} \| Q \mid \pi_X)$ is convex for fixed $P_{Y|X}$.
- This allows the use of convex optimization tools when solving for the optimal Q in minimax redundancy formulations.
- The convexity arises directly from the joint convexity of $D(P \| Q)$ and the linearity of expectation over π_X.

(20250222#119)

Prove these: For fixed $P_{Y|X}$, the function

$$\pi_X \mapsto I(X;Y)$$

is **concave** in π_X. For fixed π_X, the function

$$P_{Y|X} \mapsto I(X;Y)$$

is **convex** in $P_{Y|X}$.

Theorem: Convexity and Concavity of Mutual Information

Let $P_{Y|X}$ be a fixed conditional distribution (channel).

- For fixed $P_{Y|X}$, the function
$$\pi_X \mapsto I(X;Y)$$
is **concave** in π_X.
- For fixed π_X, the function
$$P_{Y|X} \mapsto I(X;Y)$$
is **convex** in $P_{Y|X}$.

Proof Outline

(a) Concavity in π_X:

We begin with the identity:

$$I(X;Y) = D(P_{Y|X} \,\|\, P_Y \mid \pi_X) = \sum_{x \in \mathcal{X}} \pi_X(x) D(P_{Y|X=x} \,\|\, P_Y),$$

where

$$P_Y(y) = \sum_{x \in \mathcal{X}} \pi_X(x) P_{Y|X}(y|x).$$

- The function $D(P \,\|\, Q)$ is jointly convex in (P, Q).
- P_Y is a linear function of π_X.
- $D(P_{Y|X} \,\|\, P_Y \mid \pi_X)$ is a composition of a convex function with a linear mapping, which results in a **concave** function of π_X.

Alternatively, observe that:

$$I(X;Y) = D(P_{Y|X} \,\|\, Q \mid \pi_X) - D(P_Y \,\|\, Q),$$

for any distribution Q. Here:

- $D(P_{Y|X} \,\|\, Q \mid \pi_X)$ is linear in π_X,
- $D(P_Y \,\|\, Q)$ is convex in π_X since P_Y is a linear function of π_X,
- Therefore, $I(X;Y)$ is the difference of a linear and convex function $\Rightarrow$ concave in π_X.

(b) Convexity in $P_{Y|X}$:

We now fix π_X, and treat $I(X;Y)$ as a function of $P_{Y|X}$. Again, recall:

$$I(X;Y) = D(P_{Y|X} \,\|\, P_Y \mid \pi_X), \quad \text{where } P_Y(y) = \sum_x \pi_X(x) P_{Y|X}(y|x).$$

- Since $D(P \,\|\, Q)$ is jointly convex in (P, Q),
- and both $P_{Y|X}$ and the induced P_Y depend linearly on $P_{Y|X}$,
- it follows that $I(X;Y)$ is convex in $P_{Y|X}$.

Additional Observations

- $Q^* = P_Y^* = \sum_i \pi_i^* P_{Y|X}(\cdot|i)$ is the optimal average distribution that minimizes average or worst-case redundancy.

- This Q^* is not necessarily an *f-divergence average* but rather an *information center* minimizing divergence.
- The quantity
$$\max_{\pi_X} I(X;Y)$$
corresponds to the **information radius**, which represents the worst-case (maximized) mutual information across all possible input distributions.

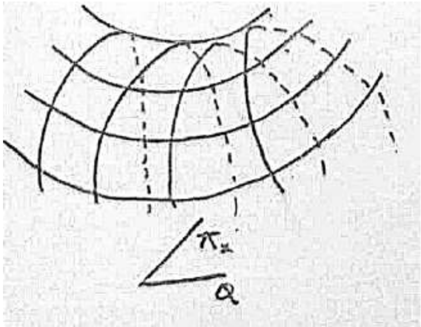
(20250222#120)

Explain saddle point inequality:

In the context of minimizing redundancy in source coding or compression, we often consider the following minimax formulation:

$$\min_{Q} \max_{\pi_X} D(P_{Y|X} \,\|\, Q \mid \pi_X),$$

which captures the worst-case average divergence (redundancy) between the actual channel $P_{Y|X}$ and an assumed output distribution Q, when the source distribution π_X is unknown and potentially adversarial. Let $Q^* = P_Y^* = \sum_x \pi_X^*(x) P_{Y|X}(\cdot|x)$ be the optimal minimizer (the information center), where π_X^* is the maximizing distribution over the input alphabet $\mathcal{X}$.



Then we have the following saddle point inequality:

$$D(P_{Y|X} \,\|\, P_Y^* \mid \pi_X) \;\leq\; D(P_{Y|X} \,\|\, P_Y^* \mid \pi_X^*) \;\leq\; D(P_{Y|X} \,\|\, Q \mid \pi_X^*),$$

which holds for any π_X and any Q.

Interpretation of the Inequalities

- The **first inequality**,
$$D(P_{Y|X} \,\|\, P_Y^* \mid \pi_X) \leq D(P_{Y|X} \,\|\, P_Y^* \mid \pi_X^*),$$
reflects the fact that, once the minimizing distribution $Q^* = P_Y^*$ is fixed, the divergence is maximized at π_X^*. That is, this is the worst-case prior under which this assumed distribution performs.

- The **second inequality**,

$$D(P_{Y|X} \| P_Y^* \mid \pi_X^*) \leq D(P_{Y|X} \| Q \mid \pi_X^*),$$

  states that for the worst-case prior π_X^*, the best assumed distribution Q is $Q^* = P_Y^*$. Any other choice of Q would yield higher average divergence.

- Together, these inequalities describe a **saddle point** (Q^*, π_X^*) of the divergence functional

$$D(P_{Y|X} \| Q \mid \pi_X),$$

  meaning that neither the compressor (choosing Q) nor the adversary (choosing π_X) can unilaterally improve their outcome by deviating from Q^* and π_X^*, respectively.

Geometric Interpretation

This saddle point inequality can also be interpreted geometrically:

- Fixing $Q = Q^*$, the divergence is maximized over π_X at π_X^*.
- Fixing $\pi_X = \pi_X^*$, the divergence is minimized over Q at Q^*.

(20250222#121)

What is conditional mutual information? Use it as a basis to explain chain rule of mutual information:

1. Conditional Mutual Information

The conditional mutual information between X and Y given Z is defined as:

$$I(X; Y \mid Z) = \mathbb{E}_{X,Y,Z}\left[\log \frac{P(X, Y \mid Z)}{P(X \mid Z)P(Y \mid Z)}\right].$$

This measures the amount of information shared between X and Y, conditioned on knowing Z.

2. Mutual Information Between a Pair and a Variable

Mutual information is symmetric with respect to grouping:

$$I((X, Y); Z) = I(X, Y; Z).$$

The mutual information between the joint variable (X, Y) and another variable Z is given by:

$$I(X, Y; Z) = \mathbb{E}_{X,Y,Z}\left[\log \frac{P(X, Y, Z)}{P(X, Y)P(Z)}\right].$$

3. Chain Rule of Mutual Information

Mutual information satisfies the following chain rule:

$$I(X, Y; Z) = I(X; Z) + I(Y; Z \mid X),$$

and similarly,

$$I(X, Y; Z) = I(Y; Z) + I(X; Z \mid Y).$$

Interpretation: The total information that the pair (X, Y) contains about Z can be broken into the information X has about Z, plus the additional information Y has about Z once X is known.

Inequality: Since conditional mutual information is always non-negative:

$$I(X, Y; Z) \geq I(X; Z), \quad I(X, Y; Z) \geq I(Y; Z).$$

4. Proof Sketch of Chain Rule

Start from:

$$I(X, Y; Z) = \mathbb{E}\left[\log \frac{P(X, Y, Z)}{P(X, Y)P(Z)}\right].$$

Factor the joint distribution:

$$\frac{P(X, Y, Z)}{P(X, Y)P(Z)} = \left(\frac{P(X, Z)}{P(X)P(Z)}\right) \cdot \left(\frac{P(Y \mid X, Z)}{P(Y \mid X)}\right).$$

Taking logarithms and expectations:

$$I(X, Y; Z) = \mathbb{E}\left[\log \frac{P(X, Z)}{P(X)P(Z)}\right] + \mathbb{E}\left[\log \frac{P(Y \mid X, Z)}{P(Y \mid X)}\right] = I(X; Z) + I(Y; Z \mid X).$$

This establishes the chain rule.

5. General Chain Rule (Multivariate)

For a sequence $X_1, X_2, \ldots, X_n$, mutual information satisfies:

$$I(X_1, X_2, \ldots, X_n; Y) = I(X_1; Y) + I(X_2; Y \mid X_1) + I(X_3; Y \mid X_1, X_2) + \cdots + I(X_n; Y \mid X_1, \ldots, X_{n-1}).$$

Proof: Follows by iterative application of the two-variable chain rule.

(20250222#122)

Explain data processing inequality:

Suppose we have a Markov chain:

$$X \to Y \to Z,$$

which implies that:

$$X \perp Z \mid Y.$$

This conditional independence tells us that once Y is known, X and Z become independent. That is, Y acts as a *sufficient statistic* of X for predicting Z.

Then, the data processing inequality states:

$$I(X;Z) \leq I(X;Y),$$

and also,

$$I(X;Z) \leq I(Y;Z).$$

Interpretation:

The idea is that if information flows from X to Z through Y, then any information Z has about X must have come via Y. Therefore, Z cannot carry more information about X than Y already does.

Proof Sketch:

We use the chain rule of mutual information:

$$I(X;Z) = I(X;Z \mid Y) + I(X;Y).$$

However, from the Markov chain assumption $X \perp Z \mid Y$, we get:

$$I(X;Z \mid Y) = 0,$$

and hence,

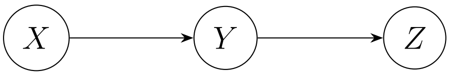
$$I(X;Z) = 0 + I(X;Y) \Rightarrow I(X;Z) \leq I(X;Y).$$

Alternatively, using the joint distribution:

$$P_{X,Z|Y} = P_{X|Y} \cdot P_{Z|Y}, \quad \text{(because } X \perp Z \mid Y\text{)},$$

which can be used to derive the inequality formally using properties of Kullback-Leibler divergence.

Illustration (Markov Chain Diagram):



Implication:

If we transform data through successive stages of processing, each stage cannot increase the amount of information the data contains about the original source.

Exercise:

Prove the inequality:

$$I(X;Z) \leq I(Y;Z),$$

given that $X \to Y \to Z$ is a Markov chain.

Hint: Use the fact that in the joint distribution, under the Markov assumption, the following always holds:

$$P_{X,Z|Y} = P_{X|Y} \cdot P_{Z|X,Y}.$$

Also remember that marginalization over joint distributions respects the Markov property:

$$P_{X,Z}(x,z) = \sum_{y} P_Y(y) P_{X|Y}(x \mid y) P_{Z|Y}(z \mid y).$$

(20250227#123)

State data processing inequality:

The **Data Processing Inequality** (DPI) states that processing data cannot increase its information content. Formally, it can be expressed as follows:

Let X_1, X_2, and X_3 be random variables forming a Markov chain denoted by $X_1 \to X_2 \to X_3$. Then, the DPI states that:

$$I(X_1; X_3) \leq I(X_1; X_2)$$

and

$$I(X_1; X_3) \leq I(X_2; X_3)$$

where $I(\cdot\,;\cdot)$ denotes the **mutual information** between two random variables.

(20250227#124)

Give the proof for data processing inequality:

We prove the DPI using the chain rule for mutual information and the Markov property.

1. **Chain Rule for Mutual Information:**

$$I(X_1; X_2, X_3) = I(X_1; X_3) + I(X_1; X_2 \mid X_3),$$

where $I(X_1; X_2 \mid X_3)$ is the conditional mutual information.

2. **Alternative Expansion:**

$$I(X_1; X_2, X_3) = I(X_1; X_2) + I(X_1; X_3 \mid X_2).$$

3. **Markov Property ($X_1 \to X_2 \to X_3$):** Since X_3 depends only on X_2 and is conditionally independent of X_1 given X_2, we have:

$$I(X_1; X_3 \mid X_2) = 0.$$

4. **Combine Results:** Equating the two expansions of $I(X_1; X_2, X_3)$:

$$I(X_1; X_2) + \underbrace{I(X_1; X_3 \mid X_2)}_{=0} = I(X_1; X_3) + I(X_1; X_2 \mid X_3).$$

Simplifying:

$$I(X_1; X_2) = I(X_1; X_3) + I(X_1; X_2 \mid X_3).$$

5. **Non-Negativity of Mutual Information:** Since $I(X_1; X_2 \mid X_3) \geq 0$, it follows that:

$$I(X_1; X_2) \geq I(X_1; X_3).$$

Equality holds if and only if $I(X_1; X_2 \mid X_3) = 0$, i.e., $X_1 \to X_3 \to X_2$ also forms a Markov chain.

(20250227#125)

Give a sufficient statistics based example for data processing inequality:

Let $X_1, \ldots, X_n$ be i.i.d. samples from an exponential family distribution with parameter θ, and let $T(X)$ be a sufficient statistic for θ. Consider the following Markov chain:

$$\theta \to X^n \to T(X) \to \hat{\theta}$$

where:

- $X^n = (X_1, \ldots, X_n)$ is the raw data,
- $T(X)$ is a sufficient statistic (e.g., sample mean for Gaussian data),
- $\hat{\theta}$ is an estimator derived from $T(X)$.

Applying the Data Processing Inequality

By the DPI, we have:

$$I(\theta; X^n) \geq I(\theta; T(X)) \geq I(\theta; \hat{\theta})$$

Interpretation

- **First Inequality ($I(\theta; X^n) \geq I(\theta; T(X))$):** The sufficient statistic $T(X)$ preserves all information about θ present in X^n, so equality holds here. This reflects the definition of sufficiency:
$$p(\theta \mid X^n) = p(\theta \mid T(X))$$

and thus:

$$I(\theta; X^n) = I(\theta; T(X))$$

- **Second Inequality ($I(\theta; T(X)) \geq I(\theta; \hat{\theta})$):** Any further processing of $T(X)$ to produce an estimate $\hat{\theta}$ cannot increase information about θ. The inequality is strict unless $\hat{\theta}$ is an invertible function of $T(X)$.

Concrete Example: Gaussian Mean Estimation

Let $X_i \sim \mathcal{N}(\theta, 1)$. Then:

- Sufficient statistic: $T(X) = \frac{1}{n}\sum_{i=1}^{n} X_i$,
- Estimator: $\hat{\theta} = T(X) + \epsilon$, where $\epsilon \sim \mathcal{N}(0, \sigma^2)$ is noise.

Here:

$$I(\theta; T(X)) = \frac{1}{2}\log(1+n), \quad I(\theta; \hat{\theta}) = \frac{1}{2}\log\left(1 + \frac{n}{1+\sigma^2}\right)$$

(20250227#126)

Explain how it is not possible to come up with single letter expressions for relay, broadcast channels:

For point-to-point channels, it is easy to come up with single letter expressions for the channel capacity. The capacity of a memoryless channel $p(y|x)$ is:

$$C = \max_{p(x)} I(X; Y)$$

This is a perfect single-letter expression.

But for many multi-user scenarios, single-letter expressions **fail to capture the capacity region**. Two classic cases:

1. Relay Channel

The capacity remains **unknown in general**, and proposed expressions require:

- Auxiliary random variables
- Non-trivial unions over distributions

- No known single-letter form

Best known achievable rate:

$$R \leq \sup \min \left\{ I(X; Y, Y_1 | X_1), I(X, X_1; Y) \right\}$$

where the supremum is over all joint distributions $p(x, x_1)$.

2. Broadcast Channel

The capacity region for general broadcast channels has:

$$\mathcal{R} = \text{conv}\left(\bigcup_{p(u,x)} \mathcal{R}(p) \right)$$

where $\mathcal{R}(p)$ is defined through:

$$R_1 \leq I(X; Y_1 | U), \quad R_2 \leq I(U; Y_2)$$

This requires auxiliary U and convex hull operations.

Counterexamples Where Single-Letter Forms Fail

1. **Relay Channel with Memory**: Even for simple cases like:

   $$Y_i = X_i + X_{i-1} + Z_i$$

   no single-letter expression exists for capacity.

2. **Interference Channel**: The Han-Kobayashi region requires:

   $$\bigcup_{p(q)p(u_1|q)p(v_1|q)p(x_1|u_1,v_1)p(u_2|q)p(v_2|q)p(x_2|u_2,v_2)} \mathcal{R}(p)$$

   with multiple auxiliary variables.

3. **Fading Channels**: With channel state information:

   $$C = \mathbb{E}\left[\max_{p(x|s)} I(X; Y | S = s) \right]$$

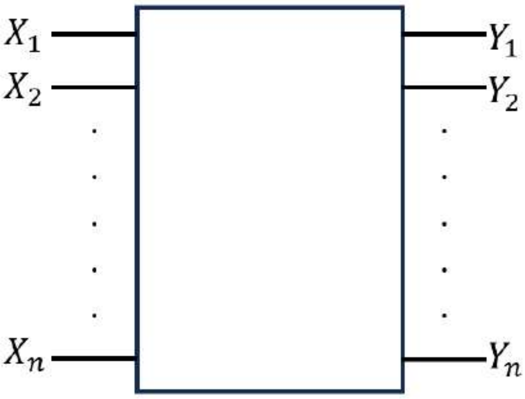
   requires expectation over states.

(20250227#127)

State the theorem connecting joint mutual information of all inputs and outputs to a channel and individual mutual information of input output pairs, given that all inputs are independent.

Theorem: Suppose $X_1, X_2, \ldots, X_n$ are independent. Then

$$I(X_1, X_2, \ldots, X_n; Y_1, Y_2, \ldots, Y_n) \geq \sum_{i=1}^{n} I(X_i, Y_i)$$



(20250227#128)

Give an example justifying the above theorem:

Suppose X_1, X_2 are i.i.d Ber(1/2).

$$Y_1 = X_1 \oplus X_2$$

$$Y_2 = Y_1 \oplus X_2 = X_1$$

What is $I(X_1; Y_1)$ and what is $I(X_2; Y_2)$?

Case	X_1	X_2	Y_1	Y_2
1	0	0	0	0
2	0	1	1	0
3	1	0	1	1
4	1	1	0	1

$$I(X_2; Y_2) = I(X_2; X_1) = 0$$

as given they are independent.

$$I(X_1; Y_1) = I(X_1; X_1 \oplus X_2)$$

$$\begin{aligned}I(X_1,X_2;Y_1,Y_2) &= H(Y_1,Y_2) - H(Y_1,Y_2|X_1,X_2)\\ &= H(X_1,X_2) - H(X_1,X_2|Y_1,Y_2)\\ &= H(X_1) + H(X_2)\\ &= 2 \text{ bits}\end{aligned}$$

The term $H(X_1, X_2|Y_1, Y_2) = 0$ because:

- From the system equations: $Y_2 = X_1$ (directly observable)
- $Y_1 = X_1 \oplus X_2$ can be combined with Y_2 to recover X_2 via:

$$X_2 = Y_2 \oplus Y_1 = X_1 \oplus (X_1 \oplus X_2)$$

- Thus (Y_1, Y_2) completely determine (X_1, X_2)

(20250227#129)

Prove $(A \oplus B) \oplus A = B$:

This result can be either obtained from the truth table, or from three key properties of XOR:

1. **Associativity**: $(x \oplus y) \oplus z = x \oplus (y \oplus z)$
2. **Commutativity**: $x \oplus y = y \oplus x$
3. **Self-inverse**: $x \oplus x = 0$

Thus:

$$(A \oplus B) \oplus A = A \oplus A \oplus B = 0 \oplus B = B$$

(20250227#130)

Prove this theorem:

$LHS - RHS =$

$$I(X_1, X_2, \ldots, X_n; Y_1, Y_2, \ldots, Y_n) - \sum_{i=1}^{n} I(X_i, Y_i)$$

$$\implies \mathbb{E}_P \log\left[\frac{P_{X_1,\ldots,X_n|Y_1,\ldots,Y_n}(X_1,\ldots,X_n|Y_1,\ldots,Y_n)}{\cancel{P_{X_1,\ldots,X_n}(X_1,\ldots,X_n)}}\frac{\cancel{\prod_{i=1}^n P_{X_i}(X_i)}}{\prod_{i=1}^n P_{X_i|Y_i}(X_i|Y_i)}\right]$$

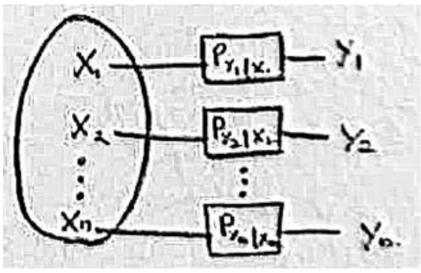
$$= D\left(P_{X_1,\ldots,X_n|Y_1,\ldots,Y_n}(X_1,\ldots,X_n|Y_1,\ldots,Y_n) \parallel \prod_{i=1}^n P_{X_i|Y_i}(X_i|Y_i) \mid P_{Y_1,\ldots,Y_n}(Y_1,\ldots,Y_n)\right) \geq 0$$

which gives

$$I(X_1, X_2, \ldots, X_n; Y_1, Y_2, \ldots, Y_n) \geq \sum_{i=1}^n I(X_i, Y_i)$$

(20250227#131)

State the theorem connecting joint mutual information of all inputs and outputs to a channel and individual mutual information of input output pairs, given that all input output pairs X_i, Y_i are independent of all other random variables.



The theorem goes as follows: Suppose given X_i, Y_i is independent of all other random variables; Then,

$$I(X_1, X_2, \ldots, X_n; Y_1, Y_2, \ldots, Y_n) \leq \sum_{i=1}^n I(X_i; Y_i)$$

Equality when given Y_i, X_i are independent of all other random variables.

(20250227#132)

Give an example showing the above theorem:

Suppose $X_1 = \text{Ber}(1/2)$ and $X_2 = X_1$, $Y_1 = X_1$ and $Y_2 = X_2$. Then,

$$\begin{aligned}I(X_1;Y_1) &= H(Y_1) - H(Y_1|X_1)\\ &= H(X_1) - H(X_1|Y_1)\\ &= 1 \text{ bit}\end{aligned}$$

$$\begin{aligned}I(X_2;Y_2) &= H(Y_2) - H(Y_2|X_2)\\ &= H(X_2) - H(X_2|Y_2)\\ &= 1\end{aligned}$$

$$\begin{aligned}I(X_1,X_2;Y_1,Y_2) &= H(X_1,X_2) - H(X_1,X_2|Y_1,Y_2)\\ &= H(Y_1,Y_2) - H(Y_1,Y_2|X_1,X_2)\\ &= H(X_1)\\ &= 1 \text{ bit}\end{aligned}$$

$$\implies I(X_1,X_2;Y_1;Y_2) \leq \sum_{i=1}^{n} I(X_i;Y_i)$$

with equality when Y_is are independent.

(20250227#133)

Prove this theorem: Given X_i, Y_i are independent of all other random variables; Then,

$$I(X_1,X_2,\ldots,X_n;Y_1,Y_2,\ldots,Y_n) \leq \sum_{i=1}^{n} I(X_i;Y_i)$$

$RHS - LHS$:

$$\begin{aligned}&\mathbb{E}\left[\log\left(\frac{\cancel{\prod_{i=1}^{n} P_{Y_i|X_i}(Y_i|X_i)}}{\prod_{i=1}^{n} P_{Y_i}(Y_i)}\frac{P_{Y_i\ldots Y_n}(Y_i\ldots Y_n)}{\cancel{P_{Y_1\ldots Y_n|X_1\ldots X_n}(Y_1\ldots Y_n|X_1\ldots X_n)}}\right]\right)\\ &= \mathbb{E}\left[\log\left(\frac{P_{Y_1\ldots Y_n}(Y_1,\ldots,Y_n)}{\prod_{i=1}^{n} P_{Y_i}}\right)\right]\\ &= D\left(P_{Y_1\ldots Y_n}(Y_1,\ldots,Y_n) \;\|\; \prod_{i=1}^{n} P_{Y_i}\right) \geq 0\end{aligned}$$

with equality when Y_is are independent.

(20250227#134)

What is a channel?

A **channel** is a mathematical model of a communication system that describes how input signals are transformed into output signals, possibly with noise or distortion. Formally:

A channel is a triple $(\mathcal{X}, \mathcal{Y}, p(y|x))$ consisting of:

- An **input alphabet** $\mathcal{X}$ (set of possible transmitted symbols)
- An **output alphabet** $\mathcal{Y}$ (set of possible received symbols)
- A **transition probability** $p(y|x)$ specifying the probability of observing output $y \in \mathcal{Y}$ given input $x \in \mathcal{X}$

(20250227#135)

When is a channel memoryless?

A channel is **memoryless** if:

$$p(y^n|x^n) = \prod_{i=1}^{n} p(y_i|x_i)$$

meaning current outputs depend only on current inputs.

(20250227#136)

Give expression for channel capacity:

The **capacity** C of a channel is the maximum achievable rate of reliable communication:

$$C = \max_{p(x)} I(X;Y)$$

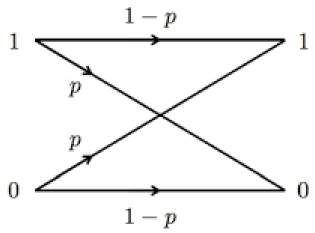
where $I(X;Y)$ is the mutual information between input and output.

(20250227#137)

Draw binary symmetric channel BSC (where the channel noise is symmetrc):

- $\mathcal{X} = \mathcal{Y} = \{0, 1\}$

- $p(0|1) = p(1|0) = \delta$ (error probability)



$$P_{Y|X}(\cdot|0) = (1 - \delta, \delta)$$
$$P_{Y|X}(\cdot|1) = (\delta, 1 - \delta)$$

Here $\delta < 1/2$.

(20250227#138)

State the channel coding theorem in simple words:

For any rate $R < C$, there exists codes with decoding error probability $\to 0$ as blocklength $\to \infty$.

(20250227#139)

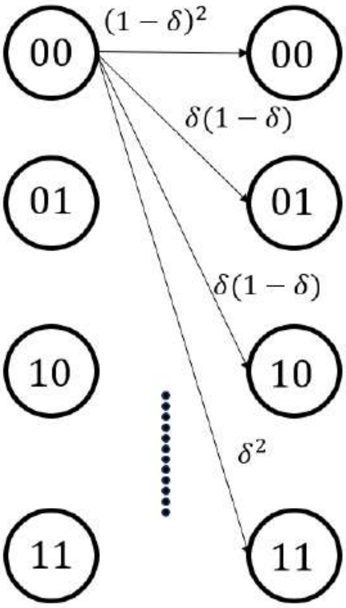
Draw binary symmetric channel with two repeated uses:

This can be generalized to n inputs scenario as well.

(20250227#140)

What is repetition coding and majority decoding?

Repetition Coding is a simple error-correcting code where each bit is repeated multiple times to provide redundancy. For an $(n, 1)$ repetition code:



- Encoding: The single information bit $b \in \{0,1\}$ is repeated n times

$$\mathrm{Enc}(b) = \underbrace{bb\cdots b}_{n \text{ times}}$$

- Rate: The code rate is $R = \frac{1}{n}$ (only 1 information bit per n coded bits)
- Example: A (3,1) repetition code encodes:

$$0 \to 000 \quad \text{and} \quad 1 \to 111$$

Majority Decoding is the corresponding decoding strategy:

- For an $(n,1)$ repetition code, the decoder examines the n received bits
- Decision rule:

$$\hat{b} = \begin{cases} 0 & \text{if received word has more 0s than 1s} \\ 1 & \text{if received word has more 1s than 0s} \end{cases}$$

- For odd n, ties cannot occur. For even n, additional rules are needed for equal counts
- Example: For (3,1) code:

$$000, 001, 010, 100 \to 0$$
$$111, 110, 101, 011 \to 1$$

The combination provides error correction capability:

- A $(3,1)$ repetition code can correct any single-bit error in each codeword
- General $(n,1)$ code can correct up to $\lfloor \frac{n-1}{2} \rfloor$ errors

- The probability of decoding error when each bit flips with probability p is:

$$P_{\text{err}} = \sum_{k=\lceil n/2 \rceil}^{n} \binom{n}{k} p^k (1-p)^{n-k}$$

Advantages	**Disadvantages**
Simple implementation	Very low code rate
Good for high-reliability channels	Inefficient for moderate noise
Useful for extreme noise cases	Better codes exist for most cases

(20250227#141)

In $(n, 1)$ repetition code, what does the 1 stand for?

In an $(n, 1)$ **repetition code**, the notation indicates:

$$(n, 1) = (\underbrace{\text{Block length}}_{n}, \underbrace{\text{Dimension}}_{1})$$

where:

- n is the **total number of bits** in each codeword (block length)
- 1 is the **number of information bits** being encoded (dimension)

This means:

- Only **1 bit** of actual information (0 or 1) is encoded
- The single bit is **repeated** n **times** to create redundancy
- Example: For $(3, 1)$ code:

$$0 \to 000 \quad \text{and} \quad 1 \to 111$$

The code's **rate** R reflects this ratio:

$$R = \frac{\text{Information bits}}{\text{Total bits}} = \frac{1}{n}$$

(20250227#142)

For the channel used n times and given input alphabet being the binary alphabet, what would be the probability of error $P_e^{(n)}$?

The probability of error for an n-repetition code with majority decoding can be analyzed as follows:

$$P_e^{(n)} = P_X(X=1)P_e^{(n)}(\text{Error} \mid X=1) + P_X(X=0)P_e^{(n)}(\text{Error} \mid X=0)$$

Let's look at the first error term:

$$P_e^{(n)}(\text{Error} \mid 0 \text{ transmitted}) \leq \Pr\left(\text{Number of 1s in output} \geq \frac{n}{2} \,\middle|\, X_1 = \cdots = X_n = 0\right)$$

The received bits $Y_1, \ldots, Y_n$ are i.i.d. Bernoulli random variables:

$$Y_i \sim \text{Ber}(\delta), \quad \text{for } i = 1, \ldots, n$$

The probability can be expressed in terms of the empirical distribution (type) τ of 1s in the output:

$$P_e^{(n)} \leq \sum_{\substack{\tau: \\ \tau(1) \geq \frac{1}{2}}} 2^{-nD(\tau\|\delta)}$$

where:

- $\tau(1)$ is the fraction of 1s in the output
- $D(\tau\|\delta)$ is the Kullback-Leibler divergence:

$$D(\tau\|\delta) = \tau(1)\log_2\frac{\tau(1)}{\delta} + (1-\tau(1))\log_2\frac{1-\tau(1)}{1-\delta}$$

This large deviations analysis shows the error probability decays exponentially with n:

$$P_e^{(n)} \approx (n+1)^{|A|}\, 2^{-nE_r(\delta)}, \quad \text{where } E_r(\delta) = \min_{\tau(1) \geq \frac{1}{2}} D(\tau\|\delta)$$

The exponent $E_r(\delta)$ is the solution to:

$$E_r(\delta) = D\left(\frac{1}{2}\|\delta\right) = 1 - H_2(\delta) - \frac{1}{2}\log_2\frac{1/4}{\delta(1-\delta)}$$

where $H_2(\delta) = -\delta\log_2\delta - (1-\delta)\log_2(1-\delta)$ is the binary entropy function.

Anyway we get,

$$P_e^{(n)} \leq (n+1)^{|A|}\, 2^{-n\min_{\tau:\tau(1)\geq 1/2} D(\tau\|\delta)}$$

and this → 0 exponentially fast as $n \to \infty$.

(20250227#143)

Find the number of bits per channel use that we can send across a channel (say a BSC) in binary hypothesis testing, multiple hypothesis case and in zero-error case:

- **Binary Hypothesis Testing:**

$$H_0 : 0 \dots 0, \quad H_1 : 1 \dots 1$$

  - Number of bits per channel use: $\frac{1}{n}$
  - Error probability decays exponentially:

$$P_e^{(n)} = 2^{-nD} \to 0 \quad \text{as } n \to \infty$$

- **Multiple Hypotheses (e.g., 4 Hypotheses):**

$$H_0, H_1, H_2, H_3 \Rightarrow 2 \text{ bits of information}$$

  - Number of bits per channel use: $\frac{2}{n}$
  - Question: Can $P_e^{(n)} \to 0$ as $n \to \infty$?

- **General Question:**

$$\text{How many hypotheses } M_n \text{ can we support such that } P_e^{(n)} \leq \varepsilon?$$

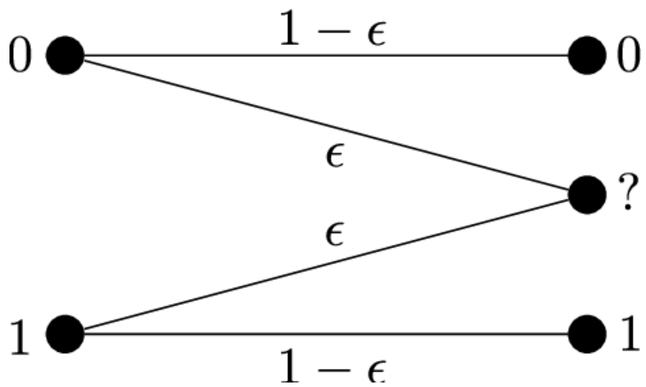
- **Zero-Error Case:**

$$\delta = 0 \Rightarrow P_e^{(n)} = 0$$

$$M_n \leq 2^n \Rightarrow \text{Bits per channel use} = \frac{\log_2 M_n}{n} = 1$$

(20250227#144)

Find the capacity of a Binary erasure channel with feedback:



Channel Model:

- Binary erasure channel with erasure probability ε.
- Noiseless feedback from receiver to transmitter.

Strategy:

- Transmit each bit until it is successfully received.
- Thanks to feedback, the transmitter knows if a bit was erased and can retransmit accordingly.

Expected Number of Channel Uses Per Bit:

The number of attempts until success follows a geometric distribution with success probability $1-\varepsilon$. Hence, the expected number of transmissions for one bit is:

$$\mathbb{E}[\#\text{ of channel uses per bit}] = \sum_{k=1}^{\infty} k \cdot (1-\varepsilon) \cdot \varepsilon^{k-1} = \frac{1}{1-\varepsilon}$$

Effective Transmission Rate:

The number of bits per channel use is given by:

$$\text{Rate} = \frac{1}{\mathbb{E}[\#\text{ of channel uses per bit}]} = 1-\varepsilon \quad \text{bits/channel use}$$

Note on Finite Hypothesis Case:

If we restrict to a finite number of hypotheses (messages), and ensure zero error by repeating transmissions as necessary, then:

$$\text{Bits per channel use} = \frac{\log_2 M}{n} \to 0 \quad \text{as } n \to \infty$$

That is, the reliable transmission rate per channel use vanishes as the blocklength grows, unless we allow for errors or use capacity-achieving strategies.

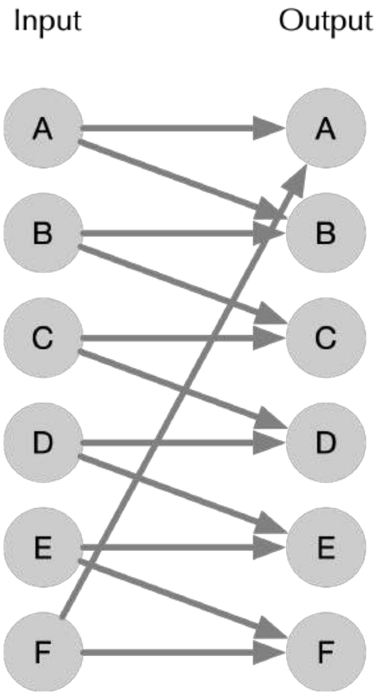
(20250227#145)

Find the channel capacity of a noisy typewriter with 5 inputs if zero-error coding assumed:

Zero-Error Communication Strategy

- We aim to find codewords such that the receiver can decode without any error.
- This requires selecting input symbols whose output sets do not overlap.

If we use the channel only once:



- We can choose two symbols, say A and C, such that their output sets are disjoint.
- This gives a zero-error code with 2 messages.
- Hence, we can send:

$$\log_2(2) = 1 \text{ bit per channel use.}$$

(20250227#146)

Give the transition probability matrices for BSC, bianry erasure channel and noisy typewriter with 5 inputs

---

1. Binary Symmetric Channel (BSC) Let the crossover probability be p. Then the transition probability matrix is:

$$P_{Y|X}^{\text{BSC}} = \begin{bmatrix} 1-p & p \\ p & 1-p \end{bmatrix}$$

Rows: input $X \in \{0,1\}$, *Columns: output* $Y \in \{0,1\}$

—

2. Binary Erasure Channel (BEC) Let the erasure probability be ε. The output alphabet is $\{0,1,e\}$, where e denotes an erasure. Then:

$$P_{Y|X}^{\mathrm{BEC}} = \begin{bmatrix} 1-\varepsilon & 0 & \varepsilon \\ 0 & 1-\varepsilon & \varepsilon \end{bmatrix}$$

Rows: input $X \in \{0,1\}$*, Columns: output* $Y \in \{0,1,e\}$

—

3. Noisy Typewriter Channel with 5 Inputs

Each input symbol outputs either itself or the next symbol (mod 5), both with probability $\frac{1}{2}$. The input and output alphabets are:

$$\{0,1,2,3,4\}$$

Transition probability matrix $P_{Y|X}^{\mathrm{NT}}$ is:

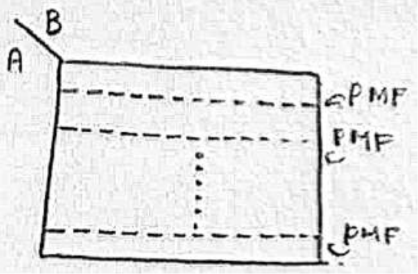
$$P_{Y|X}^{\mathrm{NT}} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & 0 & 0 & 0 & \frac{1}{2} \end{bmatrix}$$

Rows: input symbols $X = 0,1,2,3,4$*; Columns: output symbols* $Y = 0,1,2,3,4$

(20250304#147)

Define discrete channel:

A discrete channel is a tuple $(A, B, P_{Y|X})$ where $|A| < \infty$, $|B| < \infty$ and $P_{Y|X}$ is a stochastic matrix (each row of that matrix corresponds to $a_1, a_2, \ldots, a_{|A|}$ and columns corresponds to $b_1, b_2, \ldots, b_{|B|}$).



(20250304#148)

What is a discrete memoryless channel?

A discrete memoryless channel (DMC) also denoted $(A, B, P_{Y|X})$ is a sequence (meaning we're using the channel n times) of discrete channels where

$$P_{Y^n|X^n}(b^n|a^n)_{n\geq 1} = \prod_{i=1}^{n} P_{Y|X}(b_i|a_i)$$

Here the transition matrix is the same for each input, but what is input into the channel can be different.

(20250304#149)

Come up with a scheme for noisy typewriter example where we can pack 2 or more bits per transition.

(20250304#150)

Give the formal definition of a discrete memoryless channel:

An (n, M_n) code of a discrete memoryless channel $(A, B, P_{Y|X})$ is made of the following:

1. A message set, $\mathbb{W}_n = \{1, 2, 3, \ldots, M_n\}$
2. An encoder $f_n : \mathbb{W}_n \to A^n$, which maps each message to an n-letter input string.
3. A code: $c_n = \{f_n(1), f_n(2), f_n(3), \ldots, f_n(M_n) \subseteq A^n$, i.e., this code just ends up being a subset of the alphabet set A^n.
4. A decoder: $\phi_n : B^n \to \mathbb{W}^n$, where the channel output is stochastic.

The output of the channel for a chosen message w will be governed by the transition probability matrix:

$$P_{Y^n|X^n}\left(b^n|f_n(w)\right) = \prod_{i=1}^{n} P_{Y|X}\left(b_i|(f_n(w))_i\right)$$

(20250304#151)

What are the performance parameters for DMCs?

An (n, M_n) code has the following properties:

1. Rate measured in number of bits transmitted per channel use.
$$\text{Rate} = \frac{\log M_n}{n} \text{ bits per channel use}$$
2. $P_{e,w}^{n,(C_n)}$ → probability of error given a message w is transmitted:
$$P_{e,w}^{n,(C_n)} = P_{Y^n|X^n}\left(\phi_n(Y^n) \neq w|f_n(w)\right)$$
We can have average probability across all the messages as well:
$$P_e^{n,(C_n)} \to \text{ average probability of error across all the messages}$$
We can either use average probability of error or maximum probability of error across all the messages. It won't affect the final capacity of the channel as we keep increasing $n \to \infty$.
$$P_e^{(n)}(c_n) = \frac{1}{M_n} \sum_{i=1}^{M_n} P_{e,w}^n$$
assuming uniform probability distribution over the message set.
$$P_e^{(n)}(c_n) = \max_{1 \leq w \leq M_n} P_{e,w}^n$$
another way of defining probability of error. This takes the maximum across all possible errors. This doesn't need to have uniform probability distribution.

(20250304#152)

It can be shown that even if we were to use average probability of error or maximum probability of error, we're going to end up with the same capacity. So doesn't that mean the capacity is agnostic to the distribution used?

(20250304#153)

When do we say that a rate R is achievable?

A rate R is achievvable if for every $0 < \epsilon < 1$, and every $\eta > 0$, $\exists$ sequence of (n, M_n) codes (indexed by $n = 1, 2, 3, \ldots$) such that the probabillity of error $P_e^n(c_n) \leq \epsilon$ for all sufficnetly large n, and

$$\frac{\log M_n}{n} \geq R - \eta$$

That is, for first few n, our rate may not be $\leq \epsilon$, but eventually it will be true no matter the value of ϵ or η.

In layman terms: achievability of rates means that we can reliably send information over a noisy channel at a certain speed (rate) with an error that becomes negligible as messages get longer. In simple terms, it tells us how fast we can communicate without losing information, even when there's noise.

(20250304#154)

Give some remarks on the achievability of rates for a DMC:

---

1. Rate 0 is achievable. This is attained by taking $M_n = 1$ for all n

   $$\frac{\log M_n}{n} = 0$$

   It is possible to get exponentially fast decay of error with $n \to \infty$.

2. If rate R is achievable, then so is any $R' \in [0, R]$, i.e., if R is achievable, then any smaller rate is also possible. In terms of rate, one may define capacity of the DMC $(A, B, P_{Y|X})$ as

   $$C = \sup\{R : R \text{ is achievable}\}$$

3. The set of achievable rates is a closed set.

   If $R_n \to_{n\to\infty} R$, R_ns are achievable, then R is achievable.

Question: Is $C \geq 0$?

Yes, the channel capacity C is always non-negative.

To understand this intuitively, consider that the number of distinct messages M_n we can send over a noisy channel grows exponentially with block length n, provided we are operating below capacity.

For example, suppose we choose a communication rate $R = 0.5$, and allow a small gap $\eta = 0.1$. Then, we require that:

$$\frac{\log M_n}{n} \geq R - \eta = 0.4$$

This implies:

$$M_n \geq 2^{0.4n}$$

So, the number of messages we can send grows exponentially with n, confirming that the rate is positive and hence $C \geq 0$.

(20250304#155)

Prove this statement: If $R_n \rightarrow_{n \rightarrow \infty} R$, R_ns are achievable, then R is achievable.

Given: A sequence of rates $\{R_n\}$ such that $R_n \to R$ as $n \to \infty$, and each R_n is achievable. This means for every n, there exists a code of blocklength n with rate R_n and error probability $\varepsilon_n \to 0$ as $n \to \infty$.

To show: R is achievable.

Proof: Fix any $\eta > 0$. Since $R_n \to R$, there exists N such that for all $n \geq N$, we have

$$|R_n - R| < \eta \quad \Rightarrow \quad R_n > R - \eta.$$

Let $n \geq N$ and consider the code achieving rate R_n with vanishing error probability $\varepsilon_n \to 0$. Since $R_n > R - \eta$, this code also achieves a rate of at least $R - \eta$, for arbitrarily small $\eta > 0$, with vanishing error.

Therefore, for any $\eta > 0$, there exists a sequence of codes of increasing blocklength n such that:

$$\frac{\log M_n}{n} \geq R - \eta, \quad \text{with } \varepsilon_n \to 0.$$

This means $R - \eta$ is achievable for all $\eta > 0$. By the definition of achievable rates, this implies that R is achievable.

□

(20250304#156)

Capacity formula definition for a DMC

The capacity of the DMC $(A, B, P_{Y|X})$ is

$$C = \max_{P_X} I(X;Y)$$

(20250304#157)

Give some remarks based on the formula for channel capacity of a discrete memoryless channel:

---

1. $0 < \epsilon < 1 \rightarrow$ We don't ask for each specific ϵ. Independence of ϵ $(\epsilon > 0)$. This means that $C(\epsilon)$ based on $R(\epsilon)$ goes to C no matter what, as $n \to \infty$.
2. Single letter characterization
3. C is concave in P_X Mutual information $I(X;Y)$ has two key properties:

   1. **Concavity in P_X for fixed $P_{Y|X}$**:

   $$I(\lambda P_X^{(1)} + (1-\lambda)P_X^{(2)}; Y) \geq \lambda I(P_X^{(1)}; Y) + (1-\lambda) I(P_X^{(2)}; Y)$$

   This holds because:

   - $I(X;Y) = H(Y) - H(Y|X)$
   - $H(Y)$ is *concave* in P_X (since entropy is concave)
   - $H(Y|X)$ is *linear* in P_X (it's an average of conditional entropies)
   - Concave − Linear = Concave

   2. **Maximum of Concave Functions is Concave**: The capacity C is the pointwise maximum of $I(X;Y)$ over all P_X:

   $$C(P_X) = \sup_{P_X} I(X;Y)$$

   The supremum of a family of concave functions remains concave.

Implications

- The concavity guarantees a **unique global maximum** (capacity-achieving distribution P_X^*)
- Enables efficient optimization (e.g., Blahut-Arimoto algorithm)
- Explains why capacity is well-defined for DMCs

$$\boxed{C \text{ is concave in } P_X \text{ because mutual information } I(X;Y) \text{ is concave in } P_X \text{ for fixed } P_{Y|X}.}$$

(20250304#158)

Will the probability simplex in the probability space corresponding to channel input alphabet A correspond to a convex hull?

Let the input alphabet be $A = \{x_1, x_2, \ldots, x_k\}$. The **probability simplex** over A is defined as:

$$\Delta^{k-1} = \left\{ (p_1, \ldots, p_k) \in \mathbb{R}^k \;\middle|\; p_i \geq 0, \; \sum_{i=1}^{k} p_i = 1 \right\}$$

This set represents all possible probability distributions over the finite alphabet A.

Now define δ_i as the point mass (Dirac distribution) at symbol x_i, i.e.,

$$\delta_i = (0, \ldots, 0, 1, 0, \ldots, 0) \in \mathbb{R}^k$$

with 1 in the i-th position.

Then the convex hull of these point masses is:

$$\operatorname{conv}(\{\delta_1, \ldots, \delta_k\}) = \left\{ \sum_{i=1}^{k} \lambda_i \delta_i \;\middle|\; \lambda_i \geq 0, \; \sum_{i=1}^{k} \lambda_i = 1 \right\}$$

This is exactly the probability simplex Δ^{k-1}, confirming that the simplex is the convex hull of the point masses over the alphabet A.

(20250304#159)

Why the maximum P_X in the definition of capacity always exists?

We consider the channel capacity defined as:

$$C = \max_{P_X} I(X;Y)$$

and aim to show that this maximum is always attained (i.e., the maximum exists) for a discrete memoryless channel with finite input and output alphabets.

1. Domain of Maximization is a Compact Set

The maximization is over all input distributions P_X on a finite input alphabet $A = \{x_1, x_2, \ldots, x_k\}$. The set of all such distributions forms the probability simplex:

$$\mathcal{P} = \left\{ (p_1, \ldots, p_k) \in \mathbb{R}^k \;\middle|\; p_i \geq 0, \; \sum_{i=1}^{k} p_i = 1 \right\}$$

This set $\mathcal{P}$ is closed and bounded in $\mathbb{R}^k$, and hence compact.

2. Mutual Information is Continuous in P_X

For a fixed channel transition probability $P_{Y|X}$, the mutual information $I(X;Y)$ is a continuous function of the input distribution P_X. In particular, it is given by:

$$I(X;Y) = H(Y) - H(Y|X)$$

where both entropy terms depend continuously on P_X. Therefore, $I(X;Y)$ is continuous on the simplex $\mathcal{P}$.

3. Extreme Value Theorem

By the **Extreme Value Theorem**:

> A continuous function on a compact set attains its maximum.

Since $I(X;Y)$ is continuous and $\mathcal{P}$ is compact, the maximum is achieved at some distribution $P_X^* \in \mathcal{P}$, i.e.,

$$C = I(X;Y)\big|_{P_X = P_X^*}$$

Conclusion

The maximum in the capacity expression exists because we are maximizing a continuous function (mutual information) over a compact set (the probability simplex). Therefore, the channel capacity C is always attained at some input distribution P_X^*.

(20250304#160)

Give the proof for Shannon's (1948) channel coding theorem:

Let $(A, B, P_{Y|X})$ be a discrete memoryless channel (DMC), where A is the input alphabet, B is the output alphabet, and $P_{Y|X}$ is the transition probability. Let C denote the channel capacity defined by:

$$C = \max_{P_X} I(X;Y),$$

where $I(X;Y)$ is the mutual information between input X and output Y.

The theorem has two parts:

1. **Achievability:** For any rate $R < C$, there exists a sequence of codes such that the probability of decoding error tends to zero as $n \to \infty$.
2. **Converse:** For any sequence of codes with vanishing error probability, the rate R must satisfy $R \leq C$.

Achievability Proof (Random Coding Argument)

Let $R < C$, and fix $\delta > 0$. Choose an input distribution P_X such that $I(X;Y) - 4\delta > R$. Define n-length codewords as follows:

- Generate $M_n = \lfloor 2^{nR} \rfloor$ codewords $x^n(1), \ldots, x^n(M_n)$, each drawn i.i.d. according to P_X.

$$\frac{\log\left(2^{nR} - 1\right)}{n} \leq \frac{\log M_n}{n} \leq \frac{\log 2^{nR}}{n} \to_{n\to\infty} R$$

- **Encoding:** Message w is encoded to codeword $x^n(w)$. Pick the codewords randomly!

$$c_n = \{x^n(1), x^n(2), \ldots, x^n(M_n)\}$$

where for each $w \in \{1, 2, \ldots, M_n\}$,

$$x_n(w) = \{x_1(w), x_2(w), \ldots, x_n(w)\}$$

and $x_i(w) \sim P_x$ i.i.d. Realization of the codebook c_n is the random variable corresponding to a random matrix of dimension $n \times M_n$.

- **Decoding:** Given received sequence Y^n, find the unique w such that $(X^n(w), Y^n)$ are jointly typical.

  Using maximum likelihood (ML) estimator for the decoder is the best option. But for academic purposes we'll look at a suboptimal decoder based on typical sets because of its ease of analysis.

  Typical set decoder $A(n, \delta)$ "jointly typical" $\subset A^n \times B^n$ if $\exists$ unique $\hat{w}$ such that $(x^n(w), y^n) \in A(n, \delta)$.

  Note: We reveal the realized c_n to both the encoder and the decoder.

  Encoder: $w \to f_n(w)$, where each w gets mapped to a column on our codebook. The channel's role is to make it difficult for the analyst to recover back the input codeword. Codebook representation:

  $$c_n = \begin{bmatrix} x_1(1) & x_1(2) & \cdots & x_1(n) \\ x_2(1) & x_2(2) & \cdots & x_2(n) \\ \vdots & \vdots & \ddots & \vdots \\ x_{M_n}(1) & x_{M_n}(2) & \cdots & x_{M_n}(n) \end{bmatrix}$$

  where each column in codebook makes up a codeword.

- Channel: $Y^n \sim P_{Y^n|X^n}(\cdot|f_n(w))$

Error Analysis: The average probability of error $P_e^{(n)}$ satisfies:

1. The probability that the correct codeword is not jointly typical with Y^n tends to 0 as $n \to \infty$.

2. The probability that any other codeword is jointly typical with Y^n is bounded by:

   $$(M-1) \cdot 2^{-n(I(X;Y)-\delta)} \leq 2^{nR} \cdot 2^{-n(I(X;Y)-\delta)}.$$

   In short, error can occur if the correct codeword doesn't fall into the jointly typical set $(x^n(w), y^n) \notin A(n, \delta)$ or error can occur if $\exists$ another $\hat{w} \neq w$ which is $(x^n(w), y^n)$ is also typical (i.e., $\in A(n, \delta)$).

If $R < I(X;Y)$, then this total error probability $P_e^{(n)} \to 0$ as $n \to \infty$. Hence, any $R < C$ is achievable.

Converse Proof

Assume a sequence of codes (M_n, n) with $M_n = 2^{nR}$ and vanishing error probability. Let X^n be the input and Y^n be the output.

By Fano's inequality:

$$H(W|Y^n) \leq 1 + P_e^{(n)} \cdot nR,$$

where W is the message index. Then,

$$H(W) = I(W;Y^n) + H(W|Y^n) \leq I(W;Y^n) + n\varepsilon_n,$$

with $\varepsilon_n \to 0$ as $n \to \infty$.

Since $W \to X^n \to Y^n$ forms a Markov chain,

$$I(W;Y^n) \leq I(X^n;Y^n) = \sum_{i=1}^{n} I(X_i;Y_i) \leq nC.$$

Hence,

$$nR \leq nC + n\varepsilon_n \Rightarrow R \leq C + \varepsilon_n.$$

As $n \to \infty$, $\varepsilon_n \to 0$, so $R \leq C$.

Conclusion

Any rate $R < C$ is achievable (achievability), and no rate $R > C$ is achievable with vanishing error (converse). Therefore, C is the maximum reliable communication rate over a DMC.

(20250304#161)

Why is choosing a random codebook not a bad idea?

Why Random Codebooks Are Not a Bad Idea (Mathematical Justification): Let C_n denote a random codebook (with M_n codewords of length n), drawn according to some codebook distribution P_{C_n}. Let $P_{e,w}^{(n)}(c_n)$ denote the probability of decoding error when message w is sent using the deterministic codebook c_n.

We begin by analyzing the average probability of error over all random codebooks:

$$\sum_{c_n} P_{C_n}(c_n) \cdot \frac{1}{M_n} \sum_{w=1}^{M_n} P_{e,w}^{(n)}(c_n) = \mathbb{E}_{C_n} \left[\frac{1}{M_n} \sum_{w=1}^{M_n} P_{e,w}^{(n)}(C_n) \right].$$

where

$$P_{C_n}(c_n) = \prod_{i=1}^{n} \prod_{w=1}^{M_n} P_X\left((c_n)_{i,w}\right)$$

This is the **expected average error probability** when the codebook is randomly selected. The steps that follow are:

- Using properties of the channel and the random code construction, we can analyze this expectation using jointly typical decoding.
- For any rate $R < I(X;Y)$, the expectation can be shown to go to zero as $n \to \infty$ (using typicality arguments and the union bound).

- Therefore, since the expected error is small, there must exist at least one particular codebook $\hat{c}_n$ for which the average error satisfies:

$$\frac{1}{M_n}\sum_{w=1}^{M_n} P_{e,w}^{(n)}(\hat{c}_n) \leq \mathbb{E}_{C_n}\left[\frac{1}{M_n}\sum_{w=1}^{M_n} P_{e,w}^{(n)}(C_n)\right].$$

  So the performance of a random codebook shows the existence of good deterministic codebooks.

Assume the codebook c_n is generated by drawing each codeword independently and identically according to a distribution P_X^n, and that the decoding rule is symmetric for all messages (e.g., based on typicality or ML decoding).

Since the channel is memoryless and all codewords are generated i.i.d., the error probability $P_{e,w}^{(n)}(c_n)$ is the same for all messages w. Thus,

$$P_{e,w}^{(n)}(c_n) = P_{e,1}^{(n)}(c_n) \quad \forall w.$$

Therefore, the average over all messages becomes:

$$\frac{1}{M_n}\sum_{w=1}^{M_n} P_{e,w}^{(n)}(c_n) = P_{e,1}^{(n)}(c_n).$$

So,

$$\sum_{c_n} P_{C_n}(c_n) \cdot \frac{1}{M_n}\sum_{w=1}^{M_n} P_{e,w}^{(n)}(c_n) = \mathbb{E}_{C_n}\left[P_{e,1}^{(n)}(c_n)\right].$$

Thus, choosing a random codebook is not only ”not a bad idea”, but also central to the proof of Shannon’s channel coding theorem.

(20250305#162)

Give a proof sketch for DMC capacity theorem:

Theorem (Capacity of a Discrete Memoryless Channel (DMC))**.** *Let $(\mathcal{A}, \mathcal{B}, P_{Y|X})$ denote a DMC. Then the channel capacity is given by:*

$$C = \max_{P_X} I(X;Y),$$

where the maximization is over all input distributions P_X on $\mathcal{A}$, and $I(X;Y)$ is the mutual information between $X \sim P_X$ and $Y \sim P_{Y|X}$.

Proof Sketch:

- Fix an input distribution P_X. Then $I(X;Y)$ becomes fixed.
- Fix a rate $R < I(X;Y) - 4\delta$ for some small $\delta > 0$.
- Define the number of messages as $M_n = \lfloor 2^{nR} \rfloor$, so that:

$$\frac{\log M_n}{n} \to R \quad \text{as } n \to \infty.$$

- Choose a random codebook C_n, viewed as a matrix:

$$C_n = [x^n(1), x^n(2), \ldots, x^n(M_n)],$$

  where each codeword $x^n(w) \in \mathcal{A}^n$ is generated i.i.d. according to P_X.
- Share this codebook with both the encoder and decoder (common randomness).
- **Encoder:** message $w \in [M_n] \mapsto x^n(w) = f_n(w)$, the w-th column of C_n.
- **Channel:** for transmitted codeword $x^n(w)$, the channel output is distributed as:

$$Y^n \sim P_{Y^n|X^n}(\cdot \mid x^n(w)) = \prod_{i=1}^{n} P_{Y|X}(\cdot \mid x_i(w)).$$

- **Decoder:** instead of using ML decoding, which is hard to analyze, use a *typical set decoder*:
  - Declare $\hat{w}$ if there is a unique $\hat{w} \in [M_n]$ such that $(x^n(\hat{w}), y^n) \in A(n, \delta)$, the jointly typical set.
  - If no such unique $\hat{w}$ exists, declare $\hat{w} = 1$ by default.

  Let us examine the expected probability of error averaged over the random codebook C_n. Let $P_e^{(n)}(C_n)$ denote the average probability of error for a given codebook C_n. Then:

$$\mathbb{E}_{C_n}[P_e^{(n)}(C_n)] = \sum_{c_n} P_{C_n}(c_n) \cdot \frac{1}{M_n} \sum_{w=1}^{M_n} P_{e,w}^{(n)}(c_n),$$

where $P_{e,w}^{(n)}(c_n)$ is the probability of decoding error when message w is transmitted using codebook c_n.

Instead of analyzing the average error across all messages, due to the symmetry of the random coding construction (i.i.d. generation of codewords), it suffices to analyze the error for a fixed message, say $w = 1$. Therefore,

$$\mathbb{E}_{C_n}[P_e^{(n)}(C_n)] = \mathbb{E}_{C_n}[P_{e,1}^{(n)}(C_n)].$$

Error Event:

The error event for message $w = 1$ occurs if either:

(i) The pair $(X^n(1), Y^n) \notin A(n, \delta)$, or

(ii) There exists some $\hat{w} > 1$ such that $(X^n(\hat{w}), Y^n) \in A(n, \delta)$.

Thus, the error probability is upper bounded as:

$$\mathbb{E}_{C_n}[P_{e,1}^{(n)}(C_n)] \leq \Pr[(x^n(1), y^n) \notin A(n, \delta)] + \Pr\left[\bigcup_{\hat{w}>1} \{(x^n(\hat{w}), y^n) \in A(n, \delta)\}\right].$$

Justification via Contrapositive:

We justify the expression for the error event by a proof by contrapositive. Suppose the decoder correctly decodes to message $\hat{w} = 1$. This occurs if and only if:

- $(x^n(1), y^n) \in A(n, \delta)$, and
- There is no other $\hat{w} > 1$ such that $(x^n(\hat{w}), y^n) \in A(n, \delta)$.

Hence, the complement of this event (i.e., an error) implies at least one of the two error conditions stated earlier.

Joint Distribution:

Note that the pair $(X^n(1), Y^n)$ is jointly distributed as:

$$X^n(1) \sim \prod_{i=1}^{n} P_X, \quad Y^n \sim \prod_{i=1}^{n} P_{Y|X}(y_i \mid x_i(1)).$$

Moreover, for $\hat{w} > 1$, the codeword $X^n(\hat{w})$ is independent of Y^n, since the codebook was generated i.i.d. and independently of the message.

This independence allows us to analyze the second term in the error probability bound using union bound and properties of typicality.

We continue the analysis of the error probability for message $w = 1$, and bound the expected error over the random codebook:

$$\mathbb{E}_{C_n}\left[P_{e,1}^{(n)}(C_n)\right] \leq \Pr\left[(X^n(1), Y^n) \notin A(n, \delta)\right] + M_n \Pr\left[(X^n(2), Y^n) \in A(n, \delta)\right].$$

Joint Distributions:

- The pair $(X^n(1), Y^n) = ((X_1(1), Y_1), \ldots, (X_n(1), Y_n))$ is i.i.d. according to
$$(X_i(1), Y_i) \sim P_{XY} := P_X P_{Y|X}.$$
- The pair $(X^n(2), Y^n) = ((X_1(2), Y_1), \ldots, (X_n(2), Y_n))$ is i.i.d. according to the product of marginals:
$$X^n(2) \sim P_X^{\otimes n}, \quad Y^n \sim P_Y^{\otimes n}, \quad \text{independently}.$$
Thus, $(X^n(2), Y^n) \sim (P_X \otimes P_Y)^{\otimes n}$.

Probability of Being Jointly Typical:

What is the probability that a pair of independent sequences $(X^n(2), Y^n) \sim P_X \otimes P_Y$ appears jointly typical with respect to the distribution P_{XY}? That is,

$$\Pr\left[(X^n(2), Y^n) \in A(n, \delta)\right] \approx 2^{-nD(P_{XY} \| P_X \otimes P_Y)} = 2^{-nI(X;Y)}.$$

Implication on the Second Term:

Therefore, the second term in the error bound becomes
$$M_n \cdot \Pr\left[(X^n(2), Y^n) \in A(n, \delta)\right] \approx M_n \cdot 2^{-nI(X;Y)}.$$

Choose $M_n = \lfloor 2^{nR} \rfloor$ for some rate $R < I(X;Y)$, then:
$$M_n \cdot 2^{-nI(X;Y)} \leq 2^{n(R - I(X;Y))} \to 0 \quad \text{as } n \to \infty.$$

This implies that the probability of error decays to zero, as desired, provided that the rate $R < I(X;Y)$. This reflects the fact that the law of large numbers (in the form of the Asymptotic Equipartition Property) ensures that typicality-based decoding becomes reliable as $n \to \infty$.

(20250305#163)

Use the notion of jointly typical sets to complete the error analysis in the achievability part of the Shannon channel coding theorem:

Let $A(n, \delta)$ denote the **jointly typical set**:

$$A(n, \delta) = \left\{ (x^n, y^n) \in \mathcal{A}^n \times \mathcal{B}^n : \begin{array}{|l} \left|-\frac{1}{n} \log P_{X^n}(x^n) - H(X)\right| \leq \delta, \\ \left|-\frac{1}{n} \log P_{Y^n}(y^n) - H(Y)\right| \leq \delta, \\ \left|-\frac{1}{n} \log P_{X^n, Y^n}(x^n, y^n) - H(X, Y)\right| \leq \delta \end{array} \right\}.$$

That is, both marginals and the joint entropy must be within δ of their expected values.

As $n \to \infty$, we have:

$$P_{X^n,Y^n}(A(n,\delta)) \to 1.$$

Since each condition holds with high probability, the intersection of all three conditions (i.e., being in the jointly typical set) also holds with high probability.

Bounding the Probability of a Wrong Codeword Being Typical:

We evaluate the probability that a wrong codeword $X^n(2)$, which is independent of Y^n, appears to be jointly typical with Y^n:

$$\Pr\left[(X^n(2), Y^n) \in A(n,\delta)\right] = \sum_{(x^n,y^n)\in A(n,\delta)} P_X^n(x^n) P_Y^n(y^n).$$

Bounding the RHS:

$$\leq |A(n,\delta)| \cdot 2^{-nH(X)+n\delta} \cdot 2^{-nH(Y)+n\delta} \leq 2^{nH(X,Y)+n\delta} \cdot 2^{-nH(X)} \cdot 2^{-nH(Y)}.$$

Therefore,

$$\Pr\left[(X^n(2), Y^n) \in A(n,\delta)\right] \leq 2^{-nI(X;Y)+3n\delta}.$$

Bounding the Expected Error Probability:

Let $M_n = \lfloor 2^{nR} \rfloor \leq 2^{nR} \leq 2^{n(I(X;Y)-4\delta)}$. Then, the expected probability of error for message 1 is:

$$\mathbb{E}_{C_n}\left[P_{e,1}^{(n)}(C_n)\right] \leq \Pr\left[(X^n(1), Y^n) \notin A(n,\delta)\right] + M_n \cdot 2^{-nI(X;Y)+3n\delta}.$$

Since the first term vanishes as $n \to \infty$, and the second term is:

$$M_n \cdot 2^{-nI(X;Y)+3n\delta} \leq 2^{n(I(X;Y)-4\delta)} \cdot 2^{-nI(X;Y)+3n\delta} = 2^{-n\delta} \to 0,$$

we conclude that:

$$\mathbb{E}_{C_n}\left[P_{e,1}^{(n)}(C_n)\right] \to 0 \quad \text{as } n \to \infty.$$

Conclusion:

Hence, there exists a sequence of codes $\{C_n\}$ such that:

$$P_e^{(n)}(C_n) \to 0 \quad \text{and} \quad \frac{\log M_n}{n} \to R \geq I(X;Y) - 4\delta \geq R - 4\eta.$$

This completes the typical set based proof for achievability in the channel coding theorem.

(20250305#164)

Conclude the achievability part of the proof of capacity of DMC invoking concavity of the mapping from P_X to $I(X;Y)$.

Since δ is arbitrary, we can make $R \to I(X;Y)$ as closely as desired. Therefore, for any input distribution P_X, the mutual information $I(X;Y)$ is *achievable.*

Now consider the mapping:

$$P_X \mapsto I(X;Y),$$

where $P_{Y|X}$ is fixed (i.e., fixed channel). This mapping is known to be **concave** in P_X. Furthermore, the set of all probability mass functions over a finite input alphabet $\mathcal{X}$ is a **compact convex set**.

Therefore, by the concavity of $I(X;Y)$ and the extreme value theorem, the supremum

$$\sup_{P_X} I(X;Y)$$

is achieved at some distribution P_{X^*}, and is thus finite and achievable.

Hence, the channel capacity

$$C = \max_{P_X} I(X;Y)$$

is achievable.

(20250305#165)

Prove the converse part of DMC capacity theorem:

Setup:

Let W_n be a random message uniformly distributed over $\{1, 2, \ldots, M_n\}$. Suppose a code is given with rate R, where

$$R = \frac{1}{n} \log M_n.$$

Assume this code has average probability of error $P_e^{(n)} \leq \epsilon$, and

$$\frac{1}{n} \log M_n \geq R - \eta,$$

for arbitrarily small $\epsilon, \eta > 0$ and sufficiently large n. Then R is said to be *achievable.*

The communication process can be described as:

$$W_n \xrightarrow{f_n} X^n \xrightarrow{\text{Channel}} Y^n \xrightarrow{\phi_n} \hat{W}_n.$$

Information-Theoretic Bound:

$$\log M_n = H(W_n) = H(W_n \mid \hat{W}_n) + I(W_n; \hat{W}_n).$$

Applying the **Data Processing Inequality**:

$$I(W_n; \hat{W}_n) \leq I(X^n; Y^n).$$

So we get:

$$\log M_n \leq H(W_n \mid \hat{W}_n) + I(X^n; Y^n).$$

Using the subadditivity of mutual information over memoryless channels:

$$I(X^n; Y^n) \leq \sum_{i=1}^{n} I(X_i; Y_i) \leq nC,$$

where $C = \max_{P_X} I(X; Y)$ is the channel capacity.

Therefore,

$$\log M_n \leq H(W_n \mid \hat{W}_n) + nC.$$

Fano's Inequality:

Let $P_e^{(n)} = \Pr\{W_n \neq \hat{W}_n\}$. Then Fano's inequality states:

$$H(W_n \mid \hat{W}_n) \leq h(P_e^{(n)}) + P_e^{(n)} \log(M_n - 1) \leq h(P_e^{(n)}) + P_e^{(n)} \log M_n,$$

where $h(\cdot)$ is the binary entropy function.

Therefore,

$$\log M_n \leq h(P_e^{(n)}) + P_e^{(n)} \log M_n + nC.$$

Rearranging:

$$\log M_n (1 - P_e^{(n)}) \leq h(P_e^{(n)}) + nC.$$

Dividing by n:

$$\frac{1}{n} \log M_n \leq \frac{h(P_e^{(n)})}{n(1 - P_e^{(n)})} + \frac{C}{1 - P_e^{(n)}}.$$

Taking $n \to \infty$ and $P_e^{(n)} \to 0$, we get:

$$R = \lim_{n \to \infty} \frac{1}{n} \log M_n \leq C.$$

Conclusion: Any achievable rate must satisfy $R \leq C$. Hence, the channel capacity is the maximum rate at which reliable communication is possible.

(20250306#166)

Give remarks on the channel coding theorem with the perspective of design space exploration:

- A *code* is a form of **design**.
- The **design space** corresponds to the set of all possible codes.
- The central question becomes: *how can we explore this space* to find good codes?

1. Exploration of the Design Space of Codes:

- The design space is explored **without explicitly constructing or identifying good codes**.
- Instead of constructive methods, the focus is on **establishing existence** of good codes.

Shannon's Approach: Random Coding

- Shannon introduced the idea of **random coding** to prove the existence of codes with small error probability at rates below capacity.
- The use of probabilistic methods enables performance analysis over ensembles of codes, rather than specific constructions.
- This approach is foundational for understanding that good codes *exist*, even if we do not know how to construct them explicitly.

(20250306#167)

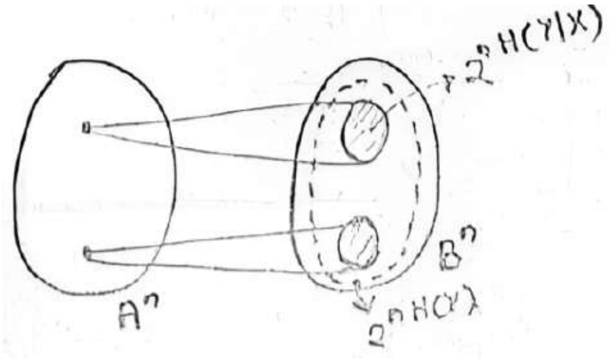
Determine the size of coneheads in the channeel output space being taken away the moment we have one code:

Cone Argument and Packing Intuition

- Once we fix a single codeword, it **occupies a cone** in the output space.
- That cone is essentially **taken away** from the rest of the design space—no other codeword can lie within it to avoid decoding ambiguity.

Packing Interpretation:

- Think of channel outputs as vectors in a high-dimensional space.



- **Code design** is analogous to **packing balls** (or cones) around each codeword such that these balls do not overlap.
- This ensures that each received vector is closest (in Hamming or another distance) to exactly one codeword.

Entropy-Based Analysis (Average Case Approximation):

- The **size of the head of a cone** (i.e., the number of typical outputs given a specific input) is approximately:
$$2^{nH(Y|X)}$$
- The **total number of typical outputs** (the effective size of the entire output space) is approximately:
$$2^{nH(Y)}$$
- Therefore, the number of cones (i.e., codewords) we can pack without overlap is:
$$\frac{2^{nH(Y)}}{2^{nH(Y|X)}} = 2^{nI(X;Y)}$$
- This matches the coding theorem's prediction: **we can reliably communicate at rates up to $I(X;Y)$.**

(20250306#168)

How does the converse of the channel coding theorem illustrate the role of mutual information as a bottleneck in reliable communication?

The converse bound for any reliable communication scheme begins with the entropy of the message:

$$\log M_n = H(W) = H(W|\hat{W}) + I(W;\hat{W})$$

This decomposition expresses the total message entropy as a sum of the residual uncertainty after decoding (error) and the mutual information between the original and decoded messages.

Using the Data Processing Inequality:

$$I(W; \hat{W}) \leq I(X^n; Y^n)$$

This follows from the Markov chain:

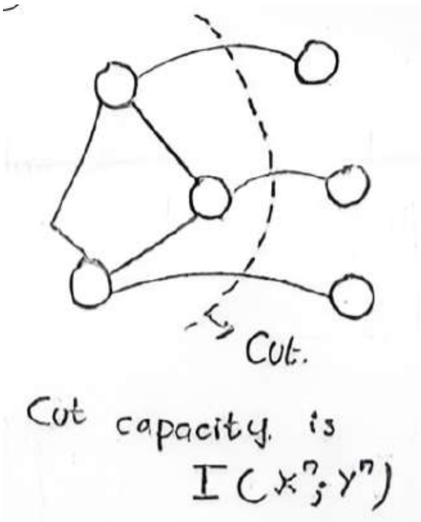
$$W \to X^n \to Y^n \to \hat{W}$$

As a result, we have:

$$\log M_n \leq H(W|\hat{W}) + I(X^n; Y^n)$$

Interpretation as a Bottleneck:

- The mutual information $I(X^n; Y^n)$ quantifies the total amount of information that can pass through the channel in n uses.
- This forms a **cut** across the channel: mutual information governs the *flow of information.*
- Hence, it acts as a **bottleneck**—no coding strategy can transmit more than what the channel itself can support.



Refining the Bound:

$$I(X^n; Y^n) \leq \sum_{i=1}^{n} I(X_i; Y_i) \leq nC$$

where C is the channel capacity per use. Thus,

$$\log M_n \leq H(W|\hat{W}) + \sum_{i=1}^{n} I(X_i; Y_i) \leq H(W|\hat{W}) + nC$$

In the limit of vanishing error probability, $H(W|\hat{W}) \to 0$, and we recover the fundamental bound:

$$\log M_n \le nC$$

Conclusion: The cut-set bound illustrates that mutual information constrains the amount of information flow through the channel, enforcing the ultimate limit on reliable communication rate.

(20250306#169)

What are some properties of good channel codes?

A good code exhibits near equality in each of the following inequalities:

$$\log M_n \le H(W|\widehat{W})+I(W;\widehat{W}) \le H(W|\widehat{W})+I(X^n;Y^n) \le H(W|\widehat{W})+\sum_{i=1}^{n} I(X_i;Y_i) \le H(W|\widehat{W})+nC$$

This chain of inequalities reflects the constraints on how much information can be reliably transmitted through a channel.

Design Implications:

- The input symbols $X_1, X_2, \ldots, X_n$ must be distributed according to the optimal input distribution P_X^*, which maximizes mutual information.
- The inputs X_i should be close to *independent* to approach equality in the step

$$I(X^n;Y^n) \le \sum_{i=1}^{n} I(X_i;Y_i).$$

(20250306#170)

How do we make the choice of typical sets in channel coding theorem?

To analyze the probability of error in the decoding process, we define a δ-typical set $A(n,\delta)$ based on the joint distribution P_{XY}. The typical set plays a central role in the decoding rule, especially when using joint typicality decoding.

The construction of $A(n, \delta)$ arises from the error analysis in the random coding argument. We must control two key events:

- $(X^n(1), Y^n)$ is jointly typical (i.e., lies in $A(n, \delta)$ with respect to P_{XY}),
- $(X^n(2), Y^n)$ is **not** jointly typical (or lies in a set typical with respect to some other joint distribution Q_{XY}).

The decoding error occurs when Y^n is jointly typical with some incorrect codeword $X^n(m)$ for $m \neq 1$. This leads us to the comparison:

$$\Pr\left[(X^n(2), Y^n) \in A(n, \delta)\right] \approx 2^{-nD(P\|Q)},$$

where:

- P is the joint distribution induced by the transmitted codeword and the channel: P_{XY},
- Q is the joint distribution when an incorrect codeword is paired with Y^n, and the two are independent: $Q_X P_Y$.

Interpretation: The event that $(X^n(2), Y^n)$ mimics the correct joint typical behavior of $(X^n(1), Y^n)$ has a probability governed by the Kullback-Leibler divergence $D(P\|Q)$. The larger this divergence, the less likely this confusion event is.

(20250306#171)

For a deterministic channel with identity mapping, obtain the channel capacity:

Consider a simple discrete memoryless channel (DMC) defined as follows:

- Input alphabet: $\mathcal{X} = \{0, 1\}$
- Output alphabet: $\mathcal{Y} = \{0, 1\}$
- Transition probabilities given by:

$$P_{Y|X} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

This channel is a **deterministic** or **noiseless** channel, where each input symbol X is mapped to the same output symbol $Y = X$ with probability 1. The system diagram is:

$$X = 0 \xrightarrow{\quad 1 \quad} Y = 0$$

$$X = 1 \xrightarrow{\quad 1 \quad} Y = 1$$

The channel capacity C is defined as:

$$C = \max_{P_X} I(X;Y)$$

We compute:

$$I(X;Y) = H(Y) - H(Y|X)$$

Step 1: Compute $H(Y|X)$. Since the channel is noiseless:

$$H(Y|X) = 0$$

Step 2: Maximize $H(Y)$ over all input distributions P_X. Since $Y = X$, the distribution of Y is exactly the same as the distribution of X, i.e., $P_Y = P_X$.

Thus, to maximize $I(X;Y)$, we need to maximize the entropy $H(Y) = H(X)$:

$$\max_{P_X} H(Y) = \max_{p\in[0,1]} H_{\text{bin}}(p)$$

where $H_{\text{bin}}(p) = -p \log p - (1-p)\log(1-p)$ is the binary entropy function.

The binary entropy is maximized when $p = 0.5$, giving:

$$\max_{P_X} H(Y) = 1$$

Conclusion: The capacity of this channel is:

$$C = \max_{P_X} I(X;Y) = \max_{P_X} H(Y) = 1$$

and this maximum is achieved when the input distribution is uniform:

$$P_X = \left(\frac{1}{2}, \frac{1}{2}\right)$$

Interpretation:

- The channel can transmit 1 bit per use without error.
- The optimal strategy is to use both input symbols with equal probability to maximize the output entropy.
- Since the mapping is one-to-one and deterministic, the output preserves all the information of the input.

(20250306#172)

For binary symmetric channel, obtain the channel capacity:

Consider the **Binary Symmetric Channel** (BSC) with crossover probability $\delta \in [0, 0.5]$.

Channel Description:

- Input alphabet: $\mathcal{X} = \{0, 1\}$
- Output alphabet: $\mathcal{Y} = \{0, 1\}$
- For each input bit $x \in \{0, 1\}$, the output is:

$$Y = \begin{cases} x & \text{with probability } 1 - \delta \\ 1 - x & \text{with probability } \delta \end{cases}$$

- Transition probability matrix:

$$P_{Y|X} = \begin{bmatrix} 1-\delta & \delta \\ \delta & 1-\delta \end{bmatrix}$$

This channel introduces noise by flipping the input bit with probability δ.

Channel Capacity:

The capacity C of this channel is given by the maximum mutual information between input and output:

$$C = \max_{P_X} I(X; Y)$$

Using the identity:

$$I(X; Y) = H(Y) - H(Y|X)$$

Step 1: Compute $H(Y|X)$.

For each fixed input x, the output is a binary random variable with bias δ or $1 - \delta$:

$$H(Y|X) = h(\delta)$$

where $h(\delta) = -\delta \log \delta - (1 - \delta) \log(1 - \delta)$ is the binary entropy function.

Step 2: Maximize $H(Y)$ over P_X.

To maximize $I(X; Y)$, we want to maximize $H(Y)$. Since Y depends on X via symmetric noise, the output entropy $H(Y)$ is maximized when $P_X = \left(\frac{1}{2}, \frac{1}{2}\right)$. In that case, Y is also uniform:

$$H(Y) = 1$$

Therefore, the mutual information becomes:

$$I(X;Y) = H(Y) - H(Y|X) = 1 - h(\delta)$$

Hence, the channel capacity is:

$$C = \max_{P_X} I(X;Y) = 1 - h(\delta)$$

Achieving the Capacity:

- The maximum is achieved when $X \sim \text{Bern}(1/2)$, i.e., $P_X = \left(\frac{1}{2}, \frac{1}{2}\right)$.
- This choice maximizes $H(Y)$ and results in the output being as unpredictable as possible (maximum entropy), given the constraint introduced by the noise level δ.

Code Design Insight:

- A **good code** for the BSC should induce input sequences X^n that are i.i.d. with $P_X = \left(\frac{1}{2}, \frac{1}{2}\right)$.
- That is, the marginal distribution of each codeword symbol should be close to uniform.
- This ensures that the typical set of codewords matches the optimal input distribution that achieves channel capacity.
- The analysis thus motivates the use of random codes where each symbol is chosen independently and uniformly at random from $\{0, 1\}$.

Conclusion:

$$\boxed{C_{\text{BSC}_\delta} = 1 - h(\delta)} \quad \text{bits per channel use}$$

and is achieved when $P_X = \left(\frac{1}{2}, \frac{1}{2}\right)$.

(20250306#173)

For a binary erasure channel with feedback and without feedback, obtain the channel capacities:

The Binary Erasure Channel (BEC_ε) has input alphabet $\mathcal{X} = \{0, 1\}$ and output alphabet $\mathcal{Y} = \{0, 1, e\}$, where e represents an erasure. The transition behavior is:

- For any input $X \in \{0, 1\}$,

$$Y = \begin{cases} X & \text{with probability } 1 - \varepsilon \\ e & \text{with probability } \varepsilon \end{cases}$$

- This means that with probability ε, the channel erases the input bit.

Capacity Derivation Without Feedback:

- The channel capacity is:

$$C = \max_{P_X} I(X;Y) = \max_{P_X} [H(Y) - H(Y|X)]$$

- Given $X = x$, the output Y is either x with probability $1 - \varepsilon$ or e with probability ε.

$$H(Y|X) = h(\varepsilon) = -\varepsilon \log \varepsilon - (1 - \varepsilon) \log(1 - \varepsilon)$$

- Now, for any input distribution $P_X = (p, 1 - p)$, the output Y is:

$$P_Y(y) = \begin{cases} p(1-\varepsilon) & \text{if } y = 0 \\ (1-p)(1-\varepsilon) & \text{if } y = 1 \\ \varepsilon & \text{if } y = e \end{cases}$$

- So,

$$H(Y) = h(\varepsilon) + (1-\varepsilon)H(X) \Rightarrow I(X;Y) = H(Y) - H(Y|X) = (1-\varepsilon)H(X)$$

- Maximized when $X \sim \text{Bern}(1/2)$:

$$H(X) = 1 \Rightarrow C = 1 - \varepsilon$$

$$\boxed{C_{\text{no feedback}} = 1 - \varepsilon}$$

Capacity Derivation With Feedback:

- Even with feedback, the receiver cannot distinguish whether the transmitted symbol was 0 or 1 if an erasure occurs.
- However, feedback allows the transmitter to know whether a bit was erased.
- The transmitter can keep resending a bit until it is successfully received.
- Thus, on average, each bit requires $1/(1 - \varepsilon)$ channel uses.
- This results in a transmission rate of $1 - \varepsilon$ bits per channel use.

$$\boxed{C_{\text{feedback}} = 1 - \varepsilon}$$

Entropy Decomposition:

- Let $E \in \{0, 1\}$ be an indicator random variable: $E = 1$ if $Y = e$ (i.e., erasure), $E = 0$ otherwise.
- Then:
$$H(Y) = H(Y, E) = H(E) + H(Y|E)$$
- We also have the identity:
$$H(Y) = H(Y, E) = H(Y) + H(E|Y) \Rightarrow H(E|Y) = 0$$
This is intuitive since once we observe Y, we know deterministically whether an erasure occurred.
- Therefore,
$$H(Y) = H(E) + H(Y|E)$$
- We can further write:
$$H(Y) = h(\varepsilon) + (1 - \varepsilon)H(Y|E = 0)$$
since Y takes value e with probability ε, and is equal to X otherwise.
- Note: When $E = 0$, the output Y reveals X, and so:
$$H(Y|E = 0) = H(X)$$
- So:
$$H(Y) = h(\varepsilon) + (1 - \varepsilon)H(X) \quad \Rightarrow \quad I(X; Y) = H(Y) - h(\varepsilon) = (1 - \varepsilon)H(X)$$

Summary:

- The entropy decomposition gives insight into how the randomness of the erasure event and the input distribution contribute to the total output entropy.
- For all inputs, the “confusion balls” (sets of inputs that can be confused due to noise) are of equal size because either the symbol is received correctly or it is completely lost.
- The fact that $H(E|Y) = 0$ reflects that the occurrence of erasure is revealed by observing the output.

(20250306#174)

For a noisy typewriter with 5 inputs, obtain the channel capacity:

This channel has input and output alphabet $\mathcal{X} = \mathcal{Y} = \{a, b, c, d, e\}$. The transition behavior is as follows:

- When a letter is transmitted, the receiver sees:

$$\text{With probability } \frac{1}{2}, \text{ the correct letter is received;}$$

$$\text{With probability } \frac{1}{2}, \text{ the next letter cyclically is received.}$$

  For example:

$$P(Y = a \mid X = a) = \frac{1}{2}, \quad P(Y = b \mid X = a) = \frac{1}{2}$$

$$P(Y = b \mid X = b) = \frac{1}{2}, \quad P(Y = c \mid X = b) = \frac{1}{2}, \quad \text{etc.}$$

- This is similar to a cyclic shift channel where each symbol can either stay the same or shift to its successor.

Capacity Derivation:

- The capacity of this channel is:

$$C = \max_{P_X} I(X;Y) = \max_{P_X} [H(Y) - H(Y|X)]$$

- Since for each input symbol, the output is either that symbol or its successor with equal probability,

$$H(Y|X) = H\left(\frac{1}{2}, \frac{1}{2}\right) = 1 \text{ bit}$$

- Hence,

$$C = \max_{P_X} H(Y) - 1$$

- The entropy $H(Y)$ is maximized when P_X is uniform:

$$P_X = \left(\frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}, \frac{1}{5}\right)$$

  In this case, Y becomes a mixture of shifted inputs and its distribution is also uniform over 5 letters.

- So,

$$H(Y) = \log 5 \quad \Rightarrow \quad C = \log 5 - 1$$

$$\boxed{C = \log 5 - 1 \approx 2.32 - 1 = 1.32 \text{ bits}}$$

Comparison to Earlier Case:

- In an earlier setup (e.g., a symmetric channel with disjoint confusion sets), we had:

$$C = \frac{1}{2} \log 5$$

- Comparing the two:

$$\log 5 - 1 > \frac{1}{2} \log 5 \quad \Leftrightarrow \quad \log 5 > 2$$

- Since:

$$\log 4 = 2 \quad \text{and} \quad \log 5 > \log 4 \Rightarrow \text{Inequality holds}$$

- Interpretation:
  - In the noisy typewriter channel, some confusion is allowed — i.e., errors are possible, but limited to two symbols.
  - In the earlier case, the decoding was made error-free, hence more stringent.
  - Allowing controlled errors increases the channel's capacity due to more flexibility in code design.

Summary:

- The noisy typewriter channel is a structured stochastic channel with fixed ambiguity (each input maps to two outputs).
- The uncertainty due to this ambiguity is exactly 1 bit, so $H(Y|X) = 1$.
- The optimal input distribution is uniform, which spreads the output entropy maximally.
- This leads to capacity:

$$C = \log 5 - 1$$

(20250306#175)

For a symmetric channel, obtain the channel capacity:

Let the input and output alphabet be

$$\mathcal{X} = \mathcal{Y} = \{0, 1, 2, \dots, r-1\}$$

Let Z be a random variable defined over the same alphabet $\mathbb{Z}_r$, with distribution:

$$\mathbb{P}(Z = i) = P_i, \quad i = 0, 1, \dots, r-1$$

The channel is defined by:

$$Y = (X + Z) \mod r$$

That is, the output is the input perturbed by Z (modulo-r addition), representing a circular symmetric noise model.

Transition Probability Matrix:

For a concrete example, suppose $r = 5$. The transition probability matrix $P_{Y|X}$ has the form:

$$P_{Y|X} = \begin{bmatrix} P_0 & P_1 & P_2 & P_3 & P_4 \\ P_4 & P_0 & P_1 & P_2 & P_3 \\ P_3 & P_4 & P_0 & P_1 & P_2 \\ P_2 & P_3 & P_4 & P_0 & P_1 \\ P_1 & P_2 & P_3 & P_4 & P_0 \end{bmatrix}$$

This matrix exhibits a **circulant structure**, where each row is a right cyclic shift of the previous row. Such a matrix represents a symmetric channel because the structure is invariant under rotation of symbols.

Capacity Derivation:

We aim to compute:

$$C = \max_{P_X} I(X;Y) = \max_{P_X} [H(Y) - H(Y|X)]$$

- Due to the channel's symmetry, the mutual information is maximized when P_X is uniform:
$$P_X(x) = \frac{1}{r}, \quad \forall x \in \mathcal{X}$$
- Under uniform P_X, the output Y also becomes uniformly distributed over $\mathcal{Y}$:
$$H(Y) = \log r$$
- The conditional entropy $H(Y|X)$ equals the entropy of the noise Z:
$$H(Y|X) = H(Z) = H(P_0, P_1, \ldots, P_{r-1})$$
because for a fixed input x, the output Y is distributed as $x + Z \mod r$, which is just a permuted version of Z and hence has the same entropy.

Therefore, the capacity is:

$$\boxed{C = \log r - H(P_0, P_1, \ldots, P_{r-1})}$$

This expression reflects that the capacity depends only on the entropy of the noise distribution Z and the size of the alphabet r.

Key Insight:

- The conditional entropy $H(Y|X)$ is independent of the input x due to the rotational symmetry.
- Hence, $I(X;Y)$ is maximized by choosing P_X uniformly.
- This is a general property of symmetric channels: uniform input maximizes mutual information.

(20250306#176)

Explain channels with feedback:

We consider a communication system where a message W_0 is transmitted over a channel with feedback. At each time i, the encoder observes the previous channel outputs $Y_1, Y_2, \dots, Y_{i-1}$ and selects the input X_i accordingly.

Communication System Structure:

- **Message:** $W_0 \in \{1, 2, \dots, M_n\}$
- **Encoder:**

$$\begin{aligned} f_1 &: W_0 \to X_1 \\ f_2 &: W_0 \times Y_1 \to X_2 \\ f_3 &: W_0 \times Y_1 \times Y_2 \to X_3 \\ &\vdots \\ f_n &: W_0 \times Y^{n-1} \to X_n \end{aligned}$$

  Each encoding function f_i adapts the input based on the message and the previous channel outputs.

- **Channel:** A discrete memoryless channel (DMC) with transition probability $P_{Y|X}$.

  The output at time i is:

$$Y_i \sim P_{Y|X}(\cdot \mid f_i(W_0, Y^{i-1}))$$

- **Decoder:** A function

$$\varphi_n : \mathcal{Y}^n \to \{1, 2, \dots, M_n\}$$

  which estimates the transmitted message from the full output sequence.

Example: In a binary symmetric channel with erasures, if the encoder knows that a previous output was an erasure, it can intelligently resend the bit. This allows the encoder to adaptively compensate for noise using feedback.

Code Definition:

An (n, M_n) code with feedback consists of:

- A message set $\{1, 2, \ldots, M_n\}$
- A sequence of encoding functions $\{f_1, f_2, \ldots, f_n\}$
- A decoding function φ_n

Performance Metrics:

- **Rate:** The rate of communication is:
$$R_n = \frac{\log M_n}{n}$$
- **Probability of Error:** For message $w \in \{1, 2, \ldots, M_n\}$, define the probability of error as:
$$P_{e,w}^{(n)} = \Pr[\varphi_n(Y^n) \neq w \mid W = w]$$
- **Maximal Probability of Error:**
$$P_e^{(n)} = \max_w P_{e,w}^{(n)}$$

Capacity with Feedback:

Define the capacity with feedback as:

$$C_F = \sup \{R : R \text{ is achievable with feedback}\}$$

It is known that for a discrete memoryless channel (DMC),

$$C = C_F$$

That is, **feedback does not increase capacity** for DMCs.

- Clearly $C \leq C_F$, since feedback gives more power to the encoder.
- However, one can ignore the feedback and use a non-feedback coding scheme that already achieves C, implying $C_F \leq C$.
- Thus, equality holds: $\boxed{C = C_F}$

Key Intuition:

- While feedback allows for adaptive transmission strategies, it does not increase the Shannon capacity of a DMC.

- Feedback may still reduce complexity, improve convergence, or reduce error probabilities, but not the asymptotic capacity.

(20250306#177)

Prove the theorem: Feedback doesn't increase the capacity of the DMC $C_F = C$ for a DMC $(A, B, P_{Y|X})$:

Let the DMC be defined by the triple $(\mathcal{A}, \mathcal{B}, P_{Y|X})$, with input alphabet $\mathcal{A}$, output alphabet $\mathcal{B}$, and channel transition probability $P_{Y|X}$.

Then, the capacity with feedback C_F satisfies:

$$C_F = C$$

Proof:

Let W be the message to be sent over the channel, and let $\hat{W}$ be the decoded message at the receiver. Let $Y^n = (Y_1, Y_2, \ldots, Y_n)$ denote the sequence of channel outputs. In the presence of feedback, the channel input X_i can depend on the message W and past outputs Y^{i-1}.

- Start with the mutual information:

  $$I(W; \hat{W}) \leq I(W; Y^n)$$

  This follows from the **data processing inequality**, since $W \to Y^n \to \hat{W}$ forms a Markov chain (decoding is based only on Y^n).

- Expand the mutual information:

  $$I(W; Y^n) = H(Y^n) - H(Y^n \mid W)$$

- Since $X_i = f_i(W, Y^{i-1})$, the channel inputs are deterministic functions of (W, Y^{i-1}), so:

  $$H(Y^n \mid W) = \sum_{i=1}^{n} H(Y_i \mid W, Y^{i-1})$$

- Now condition further on X_i, which is a function of (W, Y^{i-1}):

  $$H(Y_i \mid W, Y^{i-1}) = H(Y_i \mid W, Y^{i-1}, X_i)$$

- Since the channel is memoryless, and Y_i depends only on X_i (not on W or Y^{i-1}) given X_i:

  $$H(Y_i \mid W, Y^{i-1}, X_i) = H(Y_i \mid X_i)$$

- Hence,
$$H(Y^n \mid W) = \sum_{i=1}^{n} H(Y_i \mid X_i)$$

- Therefore,
$$I(W; Y^n) = H(Y^n) - \sum_{i=1}^{n} H(Y_i \mid X_i)$$

- Now use the chain rule:
$$H(Y^n) = \sum_{i=1}^{n} H(Y_i \mid Y^{i-1})$$

- So:
$$I(W; Y^n) = \sum_{i=1}^{n} \left[H(Y_i \mid Y^{i-1}) - H(Y_i \mid X_i) \right]$$

- Note that:
$$H(Y_i \mid Y^{i-1}) \leq H(Y_i)$$

- Thus,
$$I(W; Y^n) \leq \sum_{i=1}^{n} [H(Y_i) - H(Y_i \mid X_i)] = \sum_{i=1}^{n} I(X_i; Y_i)$$

- So:
$$\frac{1}{n} I(W; Y^n) \leq \frac{1}{n} \sum_{i=1}^{n} I(X_i; Y_i)$$

- This quantity is maximized when all X_i are i.i.d. and independent of feedback, which corresponds to the standard no-feedback setting.

- Hence, the best achievable rate with feedback is bounded above by the no-feedback capacity C:
$$C_F \leq C$$

- Since the encoder can always ignore the feedback and use the optimal no-feedback strategy, we have:
$$C \leq C_F$$

- Therefore, we conclude:
$$\boxed{C = C_F}$$

Conclusion: Feedback does not increase the capacity of a DMC. It may help in other aspects like reducing the probability of error or simplifying coding strategies, but the capacity remains the same.

2 20250312

(20250312#178)

Briefly describe channels with states:

- Constructive and destructive interference making channels go up and down → higher signals vs lower signals → larger gain vs lower gain → depends on noise in the system → signal to noise (SNR) ratio.
- Data and pilot go through the same channel.
- Erroneous, faults in CDs (troughs).

Channel is governed by states →

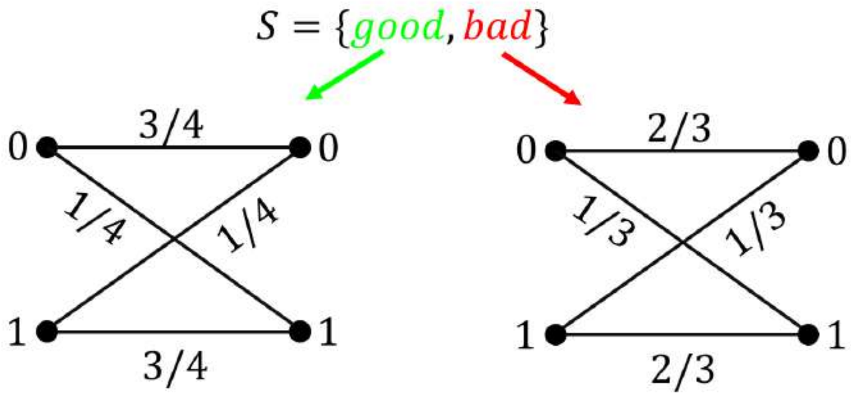
$s_1, s_2, \ldots\ iid \to P_S$ a distribution in $\mathbb{S}$

Channel: $P_{Y|X,S(b|a,s)}$, where $b \in \mathbb{B}, a \in \mathbb{A}, s \in \mathbb{S}$

Depending on the state that I'm in, the channel that I have will be different.

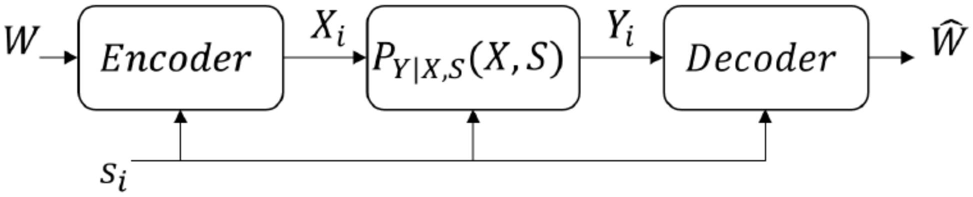
Cases: Both encoder and the decoder have access to the states. That means the Channel State Information (CSI) is available to both the transmitter/encoder and the receiver/decoder. We call such a case as CSI(TR).

Example 1 (CSITR):



In all the scenarios considered in this section, block source coding is assumed to be used.

Example 2 (CSIR): Channel state information may be available only at the receiver end {like in the pilot scenario}. Then only decoder has access to states → CSIR.



Example 3 (CSIT): Only transmitter or encoder has access to states → CSIT → **Dirty Paper Coding**.

Eg: Time Divsion Duplex systems (TDD)

Class discussion on examples for these scenarios → *Uplink, downlink, base-station (across for uplink and downlink), pilot, beamforming at the base station ??*

Eg 1: Base station sending signal to two people → Transmission to the second person treated as corrupted relative to the transmission to the first person → become a two state channel

Eg 2: Me writing on a paper. Me → Transmitter, paper → medium/channel. Paper can have smudges or not. And I'm writing on that paper. The receiver sees the messages and smudges. But the receiver doesn't know if I created those smudges or those were already part of the paper (noise in the channel) → example for CSIT.

Definition:

- C_{CSI} is the capacity under $CSI(TR)$.
- C_{CSIR} is the capacity under $CSIR$
- C_{CSIT} is the capacity under $CSIT$
- $C_{No\ CSI}$ is the capacity under $No\ \ CSI$

Suppose we have a channel with a jammer which can change the channel in whichever way it wants such that neither the transmitter nor the receiver knows the current state or state distribution → example for an arbitrary state channel or No CSI channel.

(20250312#179)

Prove these:

$$C_{CSI} = \sup_{P_{X|S}} I(X;Y|S)$$

and

$$C_{CSIR} = \sup_{P_X} I(X;Y|S)$$

Theorem:

$$(a)\ C_{CSI} = \sup_{P_{X|S}} I(X;Y|S)$$

$$(b)\ C_{CSIR} = \sup_{P_X} I(X;Y|S)$$

Proof:

(b) CSIR case:

View S as an addon to Y.

Replace Y by (Y,S).

I know $P_{Y|X,S}, P_S$. Use this to get

$$P_{Y,S|X} = P_{S|X}P_{Y|X,S} = P_S P_{Y|X,S}$$

Enhanced channel: $P_{Y,S|X} = P_S P_{Y|X,S}$

Therefore

$$\begin{aligned} C_{CSIR} &= \sup_{P_X} I\left(X;(Y,S)\right) \\ &= \sup_{P_X} \cancelto{0}{I\left(X;S\right)} + I\left(X;Y|S\right) \\ &\quad \text{(because } X \perp\!\!\!\perp S\text{)} \\ &= \sup_{P_X} I(X;Y|S) \end{aligned}$$

(a) CSI(TR) case:

$s_1, s_2, \ldots$ *iid* P_S.

Pick $r \in \mathbb{S}$.

Take n large.

Grouping all instances where channel state is r, without loss of generality, $P_S(r) > 0 \ \ \forall r \in \mathbb{S}$.

When the state is r', we have a DMC $(A, B, P_{Y|X,S}(\cdot|\cdot, S = r)$ with capacity

$$\max_{P_{X|S=r}(\cdot|r)} I(X;Y|S=r) = c_r$$

ϵ, η fixed.

For each $r \in \mathbb{S}$, $\exists \ (n, M_n^{(r)})$ block codes with $\log M_n^{(r)}/n \geq c_r - \eta$.

$P_e^n(r) \leq_n \epsilon$

$$Pr\{\frac{\#r}{n}\} \rightarrow Pr\left\{|\tau(r; S^n) - P_S(r)| > \delta\right\} \leq_n \epsilon$$

where $\#r$ denotes the number of occurrences of r in n. The above relation holds true $\forall\ r$.

$$Pr\left\{\exists\ r \in \mathbb{S} : |\tau(r; S^n) - P_S(r)| > \delta\right\} \leq_n |\mathbb{S}^n|\epsilon$$

For finite states, this works. But for ∞ states, we require a different approach.

Declare an "error" whenever this event $|\tau(r; S^n) - P_S(r)| > \delta$ happens.

Outside this event, we have

$$n\left(P_S(r) - \delta\right) \leq n\tau(r; S^n) \leq n\left(P_S(r) + \delta\right)$$

Here we choose δ such that it is smaller than the smallest of $P_S(r)$.

Thus $n\tau(r; S^n) \geq n\left(P_S(r) - \delta\right) \rightarrow \infty$ as $n \rightarrow \infty$.

(20250312#180)

Prove the maximum achievable rate when state information present:

Consider a discrete memoryless channel (DMC) where the channel state varies over time. Let the channel state sequence be denoted by $S^n = (S_1, S_2, \ldots, S_n)$, where each $S_i \in \mathbb{S}$, and suppose P_S is the distribution of the state.

Let $\tau(r; S^n)$ denote the empirical frequency of state $r \in \mathbb{S}$ in the state sequence S^n. Then:

$$n\tau(r; S^n) \geq n(P_S(r) - \delta) \geq n_r,$$

for all $r \in \mathbb{S}$, provided δ is small and n is large. This ensures that, for each state r, a subcode of blocklength n_r can be designed with rate close to the per-state capacity $C^{(r)}$:

$$\frac{1}{n_r} \log M_{n_r}^{(r)} \geq C^{(r)} - \eta, \quad P_e^{(n)}(r) \leq \epsilon.$$

Construction of Total Code:

When the state is r, the number of messages that can be sent is $M^{(r)}_{n\tau(r;S^n)}$. The total number of messages in this scheme is:

$$M_n = \prod_{r\in\mathbb{S}} M^{(r)}_{n\tau(r;S^n)}.$$

Hence, the overall rate is:

$$\frac{1}{n}\log M_n = \frac{1}{n}\sum_{r\in\mathbb{S}} \log M^{(r)}_{n\tau(r;S^n)} = \sum_{r\in\mathbb{S}} \tau(r;S^n)\cdot\frac{1}{n\tau(r;S^n)}\log M^{(r)}_{n\tau(r;S^n)}.$$

Using the lower bound on each per-state rate:

$$\frac{1}{n}\log M_n \geq \sum_{r\in\mathbb{S}} \tau(r;S^n)(C^{(r)}-\eta).$$

This expression resembles the effective capacity when the channel behaves as a time-division duplex (TDD) system across independent subchannels corresponding to different states.

Generalization to State-Dependent Channels:

We can show that:

$$\sum_{r\in\mathbb{S}} \tau(r;S^n)(C^{(r)}-\eta) \geq \max_{P_{X|S}} I(X;Y|S) - \eta - \delta|\mathbb{S}|\log|\mathbb{B}|,$$

where $\mathbb{B}$ is the output alphabet and $C^{(r)} = \max_{P_{X|S=r}} I(X;Y|S=r)$. Choosing δ, η sufficiently small, the extra terms can be made arbitrarily small, showing the achievability of rates up to:

$$\max_{P_{X|S}} I(X;Y|S).$$

Probability of Error:

The total probability of error can be bounded as:

$$P_e^{(n)} \leq \Pr\left\{|\tau(r;S^n) - P_S(r)| > \delta \text{ for some } r\right\} + \sum_{r\in\mathbb{S}} P_e^{(n\tau(r;S^n))}(r).$$

Using the Law of Large Numbers and standard error bounds:

$$P_e^{(n)} \leq |\mathbb{S}|\epsilon + |\mathbb{S}|\epsilon = 2|\mathbb{S}|\epsilon,$$

which goes to zero as $\epsilon \to 0$.

Converse Idea:

Apply Fano's inequality conditioned on the state sequence:

$$H(W \mid S^n) \leq H(W \mid \hat{W}, S^n) + I(W;\hat{W} \mid S^n).$$

Then use the Data Processing Inequality:

$$I(W;\hat{W} \mid S^n) \leq I(X^n;Y^n \mid S^n),$$

and bound this using the per-state capacities as:

$$I(X^n;Y^n \mid S^n) \leq \sum_{i=1}^{n} C^{(S_i)} \approx n \sum_{r\in\mathbb{S}} \tau(r;S^n)C^{(r)}.$$

Thus, the maximum achievable rate is:

$$\limsup_{n\to\infty} \frac{1}{n} \log M_n \leq \max_{P_{X|S}} I(X;Y|S).$$

(20250313#181)

Give proof for channel capacity of CSI(TR) - Converse:

Using Fano's inequality,

$$\log M_n = H(W) = H(W|S^n)$$

because for a uniformly distributed message set $\{1, 2, \ldots, M_n\}$, the total number of bits to resolve the uncertainty regarding the message is $H(W) = \log M_n$. Also since the message doesn't depend on the channel state, uncertainty pertaining to the message picked from a message set will be independent of the state of the channel, hence $H(W) = H(W|S^n)$.

Using one of the interpretations of mutual information, i.e. $I(A; B) = H(A) - H(A|B)$, we have $H(A) = H(A|B) + I(A; B)$. Replacing A with $\widehat{W}$ and B with W and conditioning the entire equation with S^n, we end up with

$$\log M_n = H(W) = H(W|\widehat{W}; S^n) + I(W; \widehat{W}|S^n).$$

The Data Processing Inequality (DPI) states that applying a deterministic function or a Markov chain transformation to a random variable cannot increase its mutual information with another variable. Decoder at the channel output end applies one such deterministic function to transform the channel output Y^n to the final decoded message $\widehat{W}$. So using DPI, one can conclude that $I(W; \widehat{W}|S^n) \leq I(W; Y^n|S^n)$. Thus,

$$H(W|\widehat{W}; S^n) + I(W; \widehat{W}|S^n) \leq H(W|\widehat{W}; S^n) + I(W; Y^n|S^n)$$

From Fano's inequality, we know that $H(W|\widehat{W} \leq h_2(p_E) + p_E \log(|M| - 1)$, where p_E is the probability of W not being equal to $\widehat{W}$ and $h_2(p_E)$ denotes the entropy associated with a Bernoulli random variable with probability of success (success here being $W \neq \widehat{W}$ being p_E. Also we know that the maximum value of $h_2(p_E) = p_E \log(1/p_E) + (1 - p_E) \log(1/(1 - p_E))$ is 1. This allows us to say

$$H(W|\widehat{W}; S^n) + I(W; Y^n|S^n) \leq 1 + p_E \log(|M_n|) + I(W; Y^n|S^n)$$

But we know $I(W; Y^n|S^n) = H(Y^n|S^n) - H(Y^n|W, S^n)$. Let's take into consideration the

RHS of this expression:

$$\begin{aligned}
H(Y^n|S^n) - H(Y^n|W, S^n) &= \sum_{i=1}^{n} H(Y_i|Y^{i-1}, S^n) - \sum_{i=1}^{n} H(Y_i|Y^{i-1}, W, S^n), \quad \text{using chain rule of entropy} \\
&\le \sum_{i=1}^{n} H(Y_i|S_i) - \sum_{i=1}^{n} H(Y_i|Y^{i-1}, W, S^n), \\
&\quad \text{because removing conditioning can only increase} \\
&\quad \text{uncertainty or keep it the same.} \\
&\le \sum_{i=1}^{n} H(Y_i|S_i) - \sum_{i=1}^{n} H(Y_i|Y^{i-1}, W, S^n, X^n), \\
&\quad \text{because } X^n = f(Y^n, S^n), \text{ so conditioning wrt } X^n \text{ doesn't change the term} \\
&= \sum_{i=1}^{n} H(Y_i|S_i) - \sum_{i=1}^{n} H(Y_i|S_i, X_i, W, Y^{i-1}, S^{i-1}, S_{i+1}^{n}) \\
&\le \sum_{i=1}^{n} H(Y_i|S_i) - \sum_{i=1}^{n} H(Y_i|S_i, X_i) \\
&= \sum_{i=1}^{n} I(X_i; Y_i|S_i) \\
&\le n \left(\max_{p_{X|S}} I(X;Y|S) \right)
\end{aligned}$$

which means

$$\frac{\log M_n}{n} \le \max_{P_{X|S}} I(X;Y|S)$$

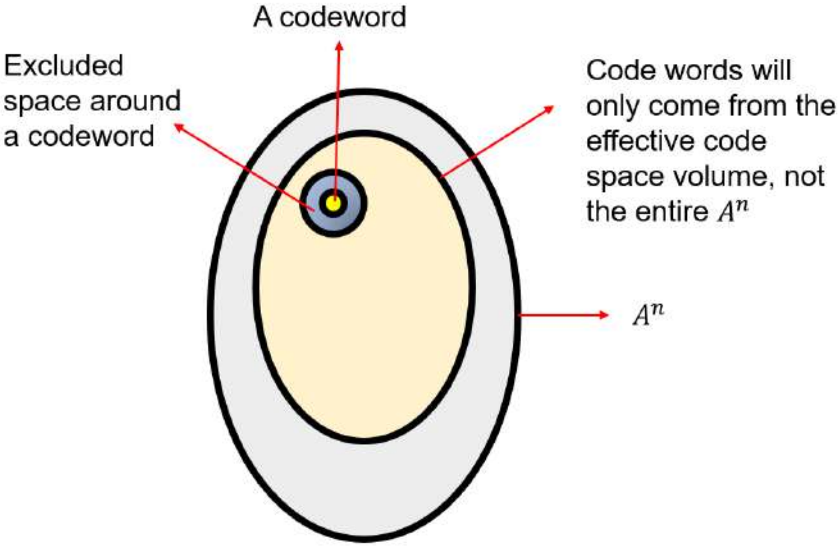
Thus from the achievability part and the converse part of the theorem, channel capacity for the CSI(TR) case will be

$$C_{CSI} = \max_{P_{X|S}} I(X;Y|S)$$

Remark: $C_{CSI} \ge C_{CSIR}$ Because in CSI case, transmitter can always choose not to use the state info and behave like a $CSIR$ at will, which means that it is possible for CSI to attain the same capacity as $CSIR$.

Mathematically, the capacity with CSITR can be expressed as an optimization problem with more degrees of freedom than the CSIR case. Since CSIR is effectively a constrained version of CSITR (where the transmitter uses a fixed strategy), the capacity with CSITR cannot be less than with CSIR. This is essentially an application of the principle that more information never hurts - having channel state information at both ends can only improve performance compared to having it at just one end.

	No CSI	CSI(TR)	CSIR	CSIT
Effective codespace volume	$2^{nH(X)}$	$2^{nH(X\|S)}$	$2^{nH(X)} = 2^{nH(X\|S)}$	$2^{nH(X\|S)}$
Excluded space around a codeword	$2^{nH(X\|Y)}$	$2^{nH(X\|Y,S)}$	$2^{nH(X\|Y,S)}$	$2^{nH(X\|Y)}$
Number of codewords	$2^{nI(X;Y)}$	$2^{nI(X;Y\|S)}$ X not $\perp\!\!\!\perp S$	$2^{nI(X;Y\|S)}$ $X \perp\!\!\!\perp S$	$2^{n(H(X\|S)-H(X\|Y))}$ $= 2^{n((I(X;Y)-I(X;S))}$


A codeword
Excluded space around a codeword
Code words will only come from the effective code space volume, not the entire A^n
A^n

In the CSIT case, the designer should exclude more space and pack the codewords more sparsely as the decoder only depends on Y and not on S. $H(X|Y,S) \leq H(X|Y) \implies$ more uncertainty is involved in this case.

(20250313#182)

Give proof sketch for Gelfand-Pinsker theorem:

$$C_{CSIT} = \sup_{P_{X,U|S}} [I(U;Y) - I(U;S)]$$

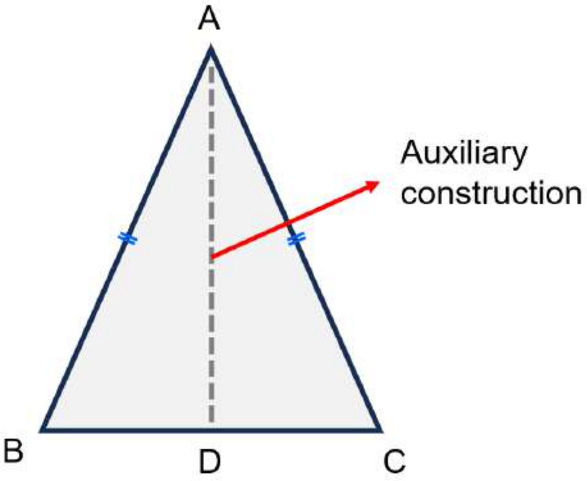
where U takes values in $\mathbb{U}$ of cardinality $\leq \max\{|A|, |B|, |S|\}$. Here $\mathbb{U}$ is the set of all auxiliary variables.

Remark 1: Something new, an auxiliary variable! Where did auxiliary construction come from?

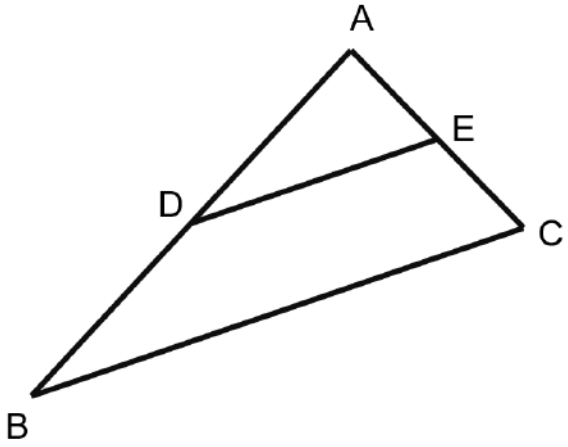
- Example 1: Auxiliary construction To prove $AB = AC$ for the isoceles triangle, we drop AD perpendicular to BC. This AD is what we call an auxiliary construction to arrive at the proof required. Depending on the problem that we require to solve, we'll have to use different auxiliary constructions.

  A strategy to solve this programmatically: Use alphageomtry (trained through reinforcement learning) to come up with an auxiliary construction, then use "Lean" to check whether $AB = AC$ can be proved with the obtained auxiliary construction, if not possible, repeat the whole process again.

- Example 2: Basic proportionality theorem



Proved using ratioing areas of the triangle ADE and triangle ABC.



Remark 2: Is $C_{CSIT} \geq I(X;Y) - I(X;S)$?

Yes, because using $U = X$ gives back the right answer (RHS). Therefore $\sup_{P_{X,U|S}}(\dots)$ gives the largest capacity.

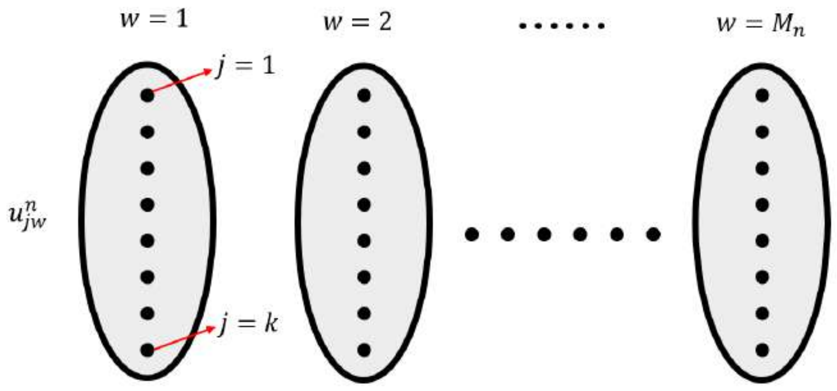
Proof idea:

Fix $\mathbb{U}$. Fix $P_{X,U|S}$. That yields a $P_U \rightarrow$ because we know P_X, P_S; use it to marginalize $P_{X,U|S}$ to get P_U.

Possibility: Encoder knows S^n, W. Pick a j such that $(U^n_{j,w}, S^n)$ is jointly typical $\rightarrow$ because it looks as if it comes from the distribution $P_{U,S}$.

Here $U^n_{j,w}$ and S^n are marginals *iid*. The probability that this $(U^n_{j,w}, S^n)$ looks like it is coming from a jointly distributed $P(U,S)$ (jointly typical) is $2^{-nI(U;S)}$ (with the obligatory $\pm\delta$, duh).

How many $u^n_{j,w}$ should I cover such that I find a $(U^n_{j,w}, S^n)$ among them which looks jointly



Each column corresponds to a bin

typical with $(U, S) \to K = 2^{nI(U;S)}$

Transmit $x^n(U_{j^*,w}, S^n) = f\left((u^n_{j,w_1}, S^n), (u^n_{j,w_2}, S^n), \ldots\right)$

It will turn out that the optimal $P_{X,U|S}$ has $x = f(U, S)$.

Decoder sees y^n. If there exists a $\hat{j}$ and $\hat{w}$ such that $\left(u^n_{\hat{j},\hat{w}}, y^n\right)$ is jointly typical typical, then declare that $\hat{w}$ as the message.

$$KM_n 2^{-nI(U;Y)} \leq_n \epsilon$$
$$M_n 2^{nI(U;S)-nI(U;Y)} \leq_n \epsilon$$

New Idea: Auxiliary construction: cloud of elements for a message

Problem Setup: Consider a channel with random state S^n known non-causally to the encoder. The channel is given by $P_{Y|X,S}$, where S^n is i.i.d. and known at the encoder. The goal is to communicate a message W over this channel.

Encoding Strategy:

- The encoder knows both the message w and the entire state sequence s^n.
- For each message w, generate $K = 2^{nI(U;S)}$ i.i.d. codewords $U^n_{j,w}$ according to the marginal P_U.
- The encoder searches for an index j^* such that $(U^n_{j^*,w}, s^n)$ is **jointly typical** with respect to the joint distribution $P_{U,S}$.

Since s^n is drawn i.i.d. from P_S, the probability that any fixed $U^n_{j,w}$ is jointly typical with s^n is approximately:

$$\mathbb{P}\left((U^n_{j,w}, s^n) \in A^{(n)}_\delta(P_{U,S})\right) \approx 2^{-nI(U;S)}$$

Hence, to ensure that at least one such j^* exists with high probability, we generate:

$$K = 2^{nI(U;S)} \quad \text{candidates per message.}$$

Transmission:

Once a suitable $U^n_{j^*,w}$ is found, transmit:

$$x^n = f(u^n, s^n) \quad \text{symbol-wise: } x_i = f(u_i, s_i)$$

where the function f is such that $X = f(U, S)$ realizes the optimal joint distribution $P_{X,U,S}$.

Decoding:

The decoder receives y^n and tries to recover w. It searches for a unique pair $(\hat{j}, \hat{w})$ such that:

$$(U^n_{\hat{j},\hat{w}}, y^n) \in A^{(n)}_\delta(P_{U,Y})$$

- If such a unique pair exists, output $\hat{w}$.
- If no such pair or multiple exist, declare an error.

Error Analysis:

To ensure small decoding error, we bound the probability that some (j, w) (with $w \neq \hat{w}$) satisfies $(U^n_{j,w}, y^n)$ jointly typical:

$$KM_n \cdot 2^{-nI(U;Y)} \leq \varepsilon$$

$$\Rightarrow M_n \leq 2^{n(I(U;Y)-I(U;S))}$$

Thus, an achievable rate is:

$$R < I(U;Y) - I(U;S)$$

Summary of Auxiliary Construction ("Cloud centers"):

- For each message w, a "cloud" of $2^{nI(U;S)}$ codewords $U^n_{j,w}$ is generated.
- Each cloud contains many candidates to match against the known state sequence s^n.
- Once a suitable codeword is found, the encoder maps it (jointly with the state) to a channel input sequence.

Conclusion:

The Gelfand–Pinsker theorem shows that with non-causal knowledge of the state at the encoder, the capacity is:

$$C = \max_{P_{U|S},\, X=f(U,S)} [I(U;Y) - I(U;S)]$$

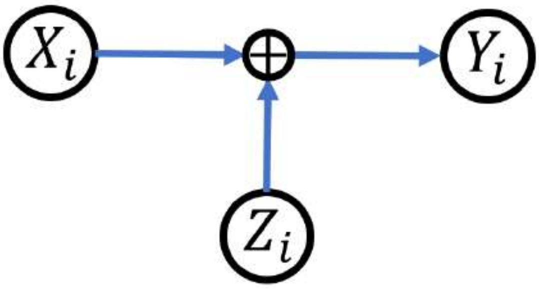
(20250325#183)

What is a gaussian channel and why is it important?

A Gaussian channel is a mathematical model for a communication channel where the output is the input plus additive Gaussian noise. It's defined as:

$$Y_i = X_i + Z_i$$

where X_i is the transmitted signal, Z_i is the noise and Y_i is the received signal at time i. The noise Z_i is typically assumed to be independent identically distributed with a Gaussian distribution, $Z_i \sim \mathcal{N}(0, \sigma^2)$



It is important because of the following reasons:

- **Physical realism**: In the physical layer (e.g., wireless, optical, or wired communication), noise from thermal effects, interference, or environmental factors often approximates a Gaussian distribution due to the Central Limit Theorem — many small, independent noise sources sum to a Gaussian profile.
- **Analytical Tractability**: Gaussian noise simplifies capacity calculations (e.g., Shannon's formula), making it a cornerstone for designing modulation, coding, and error correction schemes.
- **Universality**: It's a worst-case noise model for continuous channels under power constraints, providing a benchmark for performance (e.g., AWGN—Additive White Gaussian Noise—channel in radio systems).
- **Relevance**: Engineers at the physical layer use it to optimize signal-to-noise ratio (SNR), bit error rates, and bandwidth efficiency, critical for systems like 5G or satellite links.

In real systems, the iid nature of noise Z_i might not necessarily hold (e.g., correlated noise in fading channels). In such scenarios, one may require extensions like autoregressive models.

(20250325#184)

How does the additive noise channel enable shifting the mean of the output, and what does it mean that the transmitter controls the mean in such channels?

In the model $Y_i = X_i + Z_i$, where $Z_i \sim \mathcal{N}(0, \sigma^2)$, we have $\mathbb{E}[Z_i] = 0$ and hence $\mathbb{E}[Y_i] = \mathbb{E}[X_i + Z_i] = \mathbb{E}[X_i] + \mathbb{E}[Z_i] = \mathbb{E}[X_i]$. If the transmitter sets $X_i = x$ (deterministic value), then $\mathbb{E}[Y_i] = x$. Thus the transmitter directly controls the mean of Y_i, while the noise adds random fluctuations around it.

The implication is in the signal to noise ratio (SNR). The signal power (related to $\mathbb{E}[X_i^2]$) as compared to noise power (σ^2) determines detectability.

(20250325#185)

How is signal power related to $\mathbb{E}[X_i^2]$?

(20250325#186)

Why is gaussian channel a foundational concept in information theory?

Few reasons why its a foundational concept:

- It has the highest differential entropy as compared with all other noise distributions with fixed variance σ^2, making it the worst-case noise for capacity under power constraints (max-min principle).
- Central limit theorem: Real world noise often converges to Gaussian due to multiple independent sources, validating the model.
- For a band-limited channel with bandwidth W and power P, the capacity is $C = W \log_2(1 + P/N_0 W)$, where $N_0/2$ is the noise power spectral density ($\sigma^2 = N_0 W$). This underpins modern communication system design.

(20250325#187)

How to prove that gaussian channels have highest differential entropy as compared with all other noise distributions with fixed variance?

(20250325#188)

What are some extensions of gaussian channels?

Some extensions include:

- Fading channels: $Y_i = h_i X_i + Z_i$, where h_i varies (e.g., Rayleigh fading), complicating the control.
- Parallel Channels: Multiple Gaussian channels (e.g., MIMO) use water-filling for power allocation.

(20250325#189)

What does the observation $\sigma^2 = 0$ imply for a gaussian channel and why does it lead to $C = \infty$?

Noise variance is 0. Then $Y_i = X_i + Z_i = X_i$, indicating a noiseless channel. Without noise, the receiver can distinguish any two X_i values perfectly, allowing an infinite number of messages to be transmitted reliably, limited only by the input range and power constraint. Here assuming $X_i \in \mathbb{R}$, the number of input messages will become ∞.

For such a channel, signal to noise ratio SNR $P/\sigma^2 \to \infty$, so $C \to (1/2)\log_2(1+\infty) \to \infty$.

(20250325#190)

Give two ways to encode messages when $\sigma^2 = 0$ for a Gaussian channel:

- Mapping every message to a unique real number
- Dense packing within power constraint (implicit second way): Within the power limit $\mathbb{E}[X_i^2] \leq P$, pack as many distinct X_i values as possible (e.g., using a dense grid like $\sqrt{P} \cdot \{-1, -0.999, \ldots, 0.999, 1\}$. As the spacing δ between values approaches 0, the number of signals $N \approx 2\sqrt{P}/\delta$ grows without bound.

Contrast: The first method is a theoretical mapping of all messages to $\mathbb{R}$, while the second is a practical encoding within P, both leveraging the noiseless condition.

(20250325#191)
How to apply PAM on a theoretical gaussian channel with $\sigma^2 = 0$?

In Pulse Amplitude Modulation (PAM), we map messages to discrete levels (e.g., {1,2,3}). With $\sigma^2 = 0$, levels can be infinitesimally placed within P.

(20250325#192)
What is the expression that shows how bandwidth W limits capacity in continuous-time channels?

Bandwidth W typically caps capacity in continuous-time channels.

$$C = W \log_2 \left(1 + P/\sigma^2 W\right)$$

(20250325#193)
What is the capacity for gaussian channels with $\sigma^2 > 0$, but with no power constraints?

∞, as $C = \frac{1}{2}\log(1 + P/\sigma^2)$ and P is unrestricted allowing capacity to be infinitely large.

(20250325#194)
What would be an encoding scheme in a gaussian channel with a finitely large σ^2?

(20250325#195)
Why would using a 16/32 bits encoding rather than 4 bits encoding make sense for a gaussian channel?

High noise variance results in Z_i having large fluctuations, reducing SNR P/σ^2 and capacity, unless P increases. To combat high noise, increase X_i's magnitude (e.g., from 1 to 100), shifting the signal's energy to high values (or "upper bits" in digital terms), making it distinguishable over noise.

No matter what the σ is, given I have access to infinite power P, I can always choose larger number of bits encoding to combat the effect of noise with variance σ^2 to make sure that capacity is ∞. Because the contribution from noise will then be relatively low, only flipping least significant bits. Our message will be encoded in most significant bits, and the least significant bits can be left out to be 0s, so flipping them anyway doesn't create an ambiguity while the decoder decodes the channel output.

(20250325#196)

Why does constraining energy used per transmission make the notion of capacity meaningful?

Limiting $\mathbb{E}(X_i^2) \leq P$ reflects real systems (e.g., transmitter power limits). Capacity becomes

$$C = \log_2\left(1 + \frac{P}{\sigma^2}\right)$$

, a finite value dependent on SNR.

To improve SNR, one can increase X_i magnitude, but it comes with larger power usage and is limited by practical limits (e.g., battery life, hardware).

(20250325#197)

How is power constraint defined?

As average energy expended across multiple transmissions

$$\frac{1}{n}\sum_{i=1}^{n} X_i^2 \leq P$$

(20250325#198)

How is noise power related to law of large numbers?

Noise's sample variance is

$$\frac{1}{n}\sum_{i=1}^{n} Z_i^2$$

. By law of large numbers, for $Z_i \sim \mathcal{N}(0, \sigma^2)$,

$$\frac{1}{n}\sum_{i=1}^{n} Z_i^2 \to \mathbb{E}[Z_i^2] = \sigma^2$$

as $n \to \infty$, meaning the empirical noise power converges to σ^2.

(20250325#199)

What is the expression for total energy used for n transmissions across a gaussian channel with power constraint P?

$\to nP$

(20250325#200)

In layman terms, what does P/σ^2 mean?

It says how much more powerful is a signal compared to noise → signal-to-noise ratio.

(20250325#201)

What are peak and power constraints in the context of a Gaussian channel?

In a Gaussian channel defined by $Y_i = X_i + Z_i$, where $Z_i \mathcal{N}(0, \sigma^2)$ is Gaussian noise, constraints are imposed on the transmitted signal X_i to reflect physical or engineering limits:

Power Constraint: This limits the average energy (or power) of the signal over n channel uses:

$$\frac{1}{n}\sum_{i=1}^{n} X_i^2 \leq P$$

, where P is the maximum allowed average power. It ensures the total energy scales with time, mimicking practical limits like battery capacity or amplifier ratings in communication.

Peak Constraint: This limits the instantaneous amplitude of the signal at every time step: $|X_i| \leq A$, where A is the maximum allowed amplitude. It enforces a hard cap on the signal's magnitude, reflecting hardware limitations (e.g., maximum voltage or current in a transmitter).

(20250325#202)

Which one is a looser constraint - peak or power, and why?

Depends on their relative values and the context.

If $P = A^2$, the power constraint allows more flexibility. For example, over 4 uses, $X_i = \{A, -A, 0, 0\}$ gives $(1/4)(A^2 + A^2) = A^2/2 < A^2$, satisfying power but not a peak constraint with $A' < A$. The averaging nature means occasional large X_i are fine if balanced by smaller ones. In a Gaussian input $X_i \sim \mathcal{N}(0, P)$, $|X_i|$ can exceed $\sqrt{P}$ (e.g., 99.7% within $3\sqrt{P}$), so a peak $A \geq 3\sqrt{P}$ is needed to match, making power looser in typical cases.

If $A^2 > P$, the peak constraint allows constant $X_i = A$ (power $= A^2$), exceeding P. For instance, $A = 2$, $P = 1$ allows $X_i = 2$ (power $= 4 > 1$), fitting peak but not power. However, this is less common, as peak constraints are usually tighter in practice.

Generally, power is typically looser because it averages over time, permitting signal bursts within P, while peak enforces a rigid cap. In high-SNR regimes ($P/^2$ large), power's flexibility yields higher capacity.

(20250325#203)

When is it better to use peak constraint instead of power constraint?

In systems with strict amplitude limits (e.g., optical fibers with maximum intensity, or amplifiers with saturation).

Peak-constrained capacity is lower than power-constrained capacity for equivalent energy (e.g., $P = A^2$). The optimal input shifts from Gaussian to a discrete or uniform distribution within $[-A, A]$, depending on A/σ. For $A^2/\sigma^2 >> 1$, it approaches the power-constrained C, but for small A/σ, it's significantly less (e.g., $C \approx \log_2(2A/\sqrt{2\pi e \sigma^2})$ for uniform input).

Peak constraints suggest modulation like Pulse Amplitude Modulation (PAM) with fixed levels (e.g., $\{-A, -A/2, 0, A/2, A\}$), unlike power's continuous Gaussian input. This reduces flexibility but aligns with hardware limits.

Peak constraints are used in bandlimited channels (e.g., Nyquist signaling) or when avoiding signal clipping is critical. In wireless, power constraints dominate due to average energy concerns.

(20250325#204)
Out of power and peak constraints, which one maximizes capacity for gaussian input in a gaussian channel?

Power constraint maximizes capacity with Gaussian inputs, reflecting ideal statistical efficiency. Peak constraint reflects real-world amplitude limits, requiring adjusted inputs and lower capacity.

(20250325#205)
How are power per symbol and symbol power different?

Symbol power is the power of an instantaneous symbol, X_i^2 rather than power per symbol, which is $\mathbb{E}[X_i^2]$.

(20250325#206)
What is error power?

Likely means the same as the variance of the noise of the gaussian channel, i.e. $\mathbb{E}[Z_i^2] = \sigma^2$.

(20250325#207)
What does it mean for each codeword to have a radius in an nn-dimensional space?

In an additive white Gaussian noise (AWGN) channel, the transmitted codeword $x^n \in \mathbb{R}^n$ can be visualized as a point in an n-dimensional Euclidean space. The set of all possible transmitted codewords lies within an n-dimensional sphere of radius r, constrained by the power limitation:

$$r \leq \sqrt{np}$$

where p is the per-symbol average power constraint. This comes from the following expression

$$\frac{1}{n}\sum_{i=1}^{n} X_i^2 \leq P$$

$$\sum_{i=1}^{n} X_i^2 \leq nP$$

Since X_i^2 lies always within nP, $\sqrt{nP}$ can be thought of as the radius of a hypersphere in n-dimensional space with origin at 0 (assuming $\mathbb{E}[X_i] = 0$).

(20250325#208)

How does noise in a gaussian channel affect the received codeword in the n-dimensional space?

Total noise energy is concentrated in a shell around a magnitude of $\sqrt{n\sigma^2}$, since the noise in a gaussian channel has 0 mean and σ^2 variance.

Thus, after the transmission through the channel, the recieved signal Y_n is given by

$$Y_n = X_n + Z_n$$

. This means the received signal lies in a shell of radius $\sqrt{n\sigma^2}$ around the **transmitted codeword**.‘

(20250325#209)

But why does the noise lie concentrated in a thin shell?

In high-dimensional spaces, the probability distribution of the noise does not spread evenly but rather concentrates in a thin shell around its expected magnitude $\sqrt{n\sigma^2}$.

This phenomenon arises from the law of large numbers and concentration of measure:

- The total noise energy is approximately $n\sigma^2$, meaning the magnitude $||Z_n||$ is around $\sqrt{n\sigma^2}$.
- The probability of deviating significantly from $\sqrt{n\sigma^2}$ is exponentially small.
- The shell thickness is approximately $\sqrt{n\sigma^2} \pm \epsilon$, where ϵ becomes smaller as n increases.

This results in most of the received noise vectors lying within a narrow annular region around the transmitted codeword.

(20250325#210)

What happens to the thickness of the shell as n increases?

As $n \to \infty$, the relative thickness of the shell shrinks. The absolute deviation ϵ remains small compared to $\sqrt{n\sigma^2}$. The fractional deviation

$$\frac{\epsilon}{\sqrt{n\sigma^2}} \to 0,$$

meaning that almost all noise vectors have nearly the same magnitude.

In essence, at large n, the noise distribution becomes concentrated on an extremely thin spherical shell, making its radial component almost deterministic while the direction remains random.

(20250325#211)

How does noise directionality behave in this high-dimensional setting?

In low dimensions, Gaussian noise may exhibit an apparent directional bias. However, in high dimensions:

- The noise direction becomes uniformly distributed across all possible orientations.
- Due to spherical symmetry, the noise does not favor any particular direction.
- The distribution of noise direction is characterized by a uniform distribution over the unit sphere in $\mathbb{R}^n$, meaning that all solid angles are equally likely.

This means that, even though the magnitude of noise is almost fixed, the direction is completely random, making decoding strategies more dependent on the geometric separation of codewords rather than on individual noise realizations.

(20250325#212)

How does understanding geometric behavior of noise help us in communication system design?

- Code Design: Since noise vectors lie in a thin shell, optimal codes should maximize separation between codewords while considering their projection onto the shell. Spherical codes are often used to leverage this structure.
- Decoding Strategies: Nearest neighbor decoding is effective since noise preserves relative distances in high dimensions. Maximum likelihood decoding assumes that received signals lie in the noise shell and selects the closest codeword.
- Error Probability and Capacity: As $n \to \infty$, errors occur primarily due to codewords being too close, rather than due to large deviations of noise.

  Shannon's random coding argument leverages these high-dimensional properties to show optimal capacity-achieving codes exist.

(20250325#213)

What is meant by the "volume of probabilistically significant areas" in Gaussian channels?

In a Gaussian channel, the transmitted codewords X^n are perturbed by additive Gaussian noise, leading to a set of possible received vectors Y^n. The volume of probabilistically important areas refers to the regions in $\mathbb{R}^n$ where received signals are likely to be found, given the channel noise model.

Since Gaussian noise is concentrated in a thin shell, the received signals are likely to be found within a bounded volume around the transmitted codeword rather than spread uniformly throughout the space.

(20250325#214)

What is an (n, M) code for a Gaussian channel with power constraint P?

An (n, M) code consists of

- Message set:
  $$\mathbb{W} = \{1, 2, \ldots, M\}$$
  where M is the number of messages that needs to be transmitted.

- Encoder function f_n
$$f_n : \mathbb{W} \to \mathbb{R}^n$$
which maps from each message w to an n-dimensional codeword $x^n(w)$.
- Power constraint P: Each codeword must satisfy the average power constraint,
$$\frac{1}{n}\sum_{i=1}^{n} x_i(w)^2 \leq P$$
- Decoder function ϕ_n
$$\phi_n : \mathbb{R}^n \to \mathbb{W}$$
which maps a received vector Y^n back to the estimated message $\hat{w}$.

(20250325#215)

How is probability of error defined for a gaussian channel?

$$P_{e,w}^{(n)} = \Pr\left(\phi_n(Y^n) \neq w \mid \mathbb{W} = w\right)$$

(20250325#216)

What is the definition of the average probability of error in a Gaussian channel?

The average probability of error in an (n, M) code for a Gaussian channel is given by:

$$P_e^{(n)} = \frac{1}{M}\sum_{i=1}^{n} P_{e,w}^{(n)}$$

This metric captures the overall likelihood that a randomly chosen codeword is incorrectly decoded.

(20250325#217)

When is a rate R considered to be achievable under a power constraint P?

A rate R is achievable under power constraint P if there exists a sequence of codes (n, M_n) satisfying

- Rate condition:
$$\frac{\log M_n}{n} \geq_n R - \eta$$
for some small $\eta > 0$, meaning that the actual transmission rate is atleast close to R.
- Error condition:
$$P_e^{(n)} \leq \epsilon$$
for arbitrarily small $\epsilon > 0$, meaning that the average probability of error can be made arbitrarily small by choosing a sufficiently large n.

In short, rate R is achievable if we can find codewords that allow transmission at rate R while keeping the error probability low.

This achievability establishes fundamental trde-off between data rate and reliability in gaussian channels.

(20250325#218)

What is the definition of the capacity of a gaussian channel?

The capacity C of a gaussian channel is defined as the supremum over all achievable rates R given a power constraint P:

$$C = \sup\{R : R \text{ is achieved with power constraint } P\}$$

This means C represents the maximum reliable communication rate at which data can be transmitted while maintaining an arbitrarily low probability of error.

(20250325#219)

In gaussian channels, how to bound the number of codewords M_n using volume arguments?

We use a sphere-packing argument to estimate the maximum number of distinguishable codewords in the output space.

- The total received signal Y^n lies within a sphere of radius $\sqrt{n(P+\sigma^2)}$ because the total energy of Y^n is
$$\sum_{i=1}^{n} Y_i^2 \approx n(P+\sigma^2)$$
- Each codeword has noise added to it, which means the received codewords must be distinguishable despite the noise.
- Noise is approximately concentrated on a thin shell of radius $\sqrt{n\sigma^2}$, so each codeword occupies a noiseball of volume equal to a sphere of radius $\sqrt{n\sigma^2}$.
- The maximum number of codewords is therefore given by the ratio of the total volume of the output space to the volume of a single noise ball
$$M_n \leq \frac{\text{Volume of sphere of radius } \sqrt{n(P+\sigma^2)}}{\text{Volume of sphere of radius} \sqrt{n\sigma^2}}$$

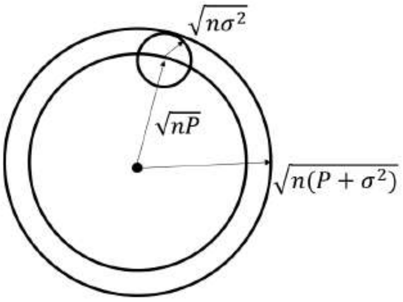
(20250325#220)

Obtain the capacity of the gaussian channel using sphere packing arguments:

Maximum number of codewords that can be reliably sent across the channel follows the relation
$$M_n \leq \frac{\text{Volume of sphere of radius } \sqrt{n(P+\sigma^2)}}{\text{Volume of sphere of radius} \sqrt{n\sigma^2}}$$
The volume of an n-dimensional sphere of radius R is proportional to
$$\text{Volume} \propto R^n$$

Thus the volume ratio simplifies to
$$M_n \leq \frac{C_n \left(n\left(P+\sigma^2\right)\right)^{n/2}}{C_n \left(n\sigma^2\right)^{n/2}} = \left(1+P/\sigma^2\right)^{n/2}.$$

Since M_n represents the number of possible messages, we conclude that M_n grows exponentially with P. Then,
$$\frac{\log M_n}{n} \leq \frac{1}{2}\log\left(1+P/\sigma^2\right)$$
which is the upper bound of achievable rate that matches with Shannon's capacity formula for the gaussian channel.
$$C = \frac{1}{2}\log\left(1+P/\sigma^2\right)$$



(20250325#221)

Can a similar argument be applied on input space?

Yes, a similar volume-packing argument applies to the input space:

- Each codeword X^n lies in a sphere of radius $\sqrt{nP}$.
- The channel introduces noise, creating overlap in the received signal space.
- By considering conditional variance (variance of X^n given Y^n), we see that the noise reduces the effective volume of the input space.

Thus, the same upper bound on M_n holds for both the output and input spaces, reinforcing the tightness of the capacity formula.

(20250325#222)

What are the major differences between DMCs and Gaussian channels?

- A Gaussian channel has a continuous input and output alphabet, whereas a discrete memoryless channel (DMC) has discrete input and output alphabets.
- Gaussian channels typically have power or peak constraints on input signals, whereas DMCs do not inherently have such constraints.

(20250325#223)

Why can't we apply the same notion of entropy that we used for discrete variables for continuous variables?

The probability of any single point in a continuous space is zero, making the standard entropy definition inapplicable. Instead, we use differential entropy, which extends the concept of entropy to continuous distributions using probability density functions (PDFs).

(20250325#224)

What is differential entropy?

Differential entropy $h(X)$ for a continuous random variable X with probability density function $f(x)$ is defined as

$$h(X) = -\int f(x) \log f(x) dx$$

or

$$h(X) = \mathbb{E}\left[\log \frac{1}{f_X(X)}\right]$$

if it exists. This measures the average uncertainty of X in terms of its probability density, analogous to entropy for discrete variables.

(20250325#225)

What is the continuous analog of minimum cardinality sets used in discrete context?

For discrete variables, we often work with typical sets, which contain most of the probability mass and have a minimum required cardinality. In continuous settings, we extend this idea by considering minimum volume sets that contain most of the probability mass.

(20250325#226)

When is a random variable X continuous?

A random variable X is continuous when its *cdf* F_X is continuous, and is the integral of its derivative f_X. i.e.,

$$F_X(x) = \int_{-\infty}^{x} f_X(u) du$$

, where

$$f_X(u) = F'_X(u)$$

for almost all x, which means the measure of sets of points where the derivative is undefined corresponds to that of a set of size 0.

(20250325#227)

Why do we require the *cdf* F_X for a random variable to be the integral of its derivative f_X for X to be deemed a continuous random variable?

Continuity of the CDF only ensures that there are no point masses (atoms) where probability is concentrated. However, it does not guarantee that the function is differentiable or that a meaningful probability density function (PDF), $f_X(x) = dF_X(x)/dx$ exists everywhere.

For example, the Cantor distribution has a continuous CDF, but no well-defined PDF because its CDF is not differentiable at many points.

(20250325#228)

Why is differential entropy called so?

The term "differential" comes from the fact that, in the continuous case, probability is described using differential elements rather than discrete probabilities.

Mathematically, it stems from the infinitesimal probability mass interpretation:

$$P(a \leq X \leq b) = \int_a^b f_X(x)dx$$

This is not just common in information theory. In game theory, when action space is continuous, we call such a space as a differential space.

(20250325#229)

Find differential entropy for a uniformly distributed random variable $X = U[0, a]$:

Let $X \sim \mathcal{U}[0, a]$. The probability density function (pdf) of X is:

$$f_X(x) = \begin{cases} \frac{1}{a}, & x \in [0, a] \\ 0, & \text{otherwise} \end{cases}$$

The differential entropy is defined as:

$$h(X) = -\int_{-\infty}^{\infty} f_X(x) \log f_X(x) \, dx.$$

Since $f_X(x) = \frac{1}{a}$ for $x \in [0, a]$, we have:

$$h(X) = -\int_0^a \frac{1}{a} \log\left(\frac{1}{a}\right) dx.$$

Note that $\log\left(\frac{1}{a}\right)$ is constant over the interval, so:

$$h(X) = -\frac{1}{a} \log\left(\frac{1}{a}\right) \int_0^a dx = -\frac{1}{a} \log\left(\frac{1}{a}\right) \cdot a = \log a.$$

Result:

$$h(X) = \log a.$$

(20250325#230)

Find differential entropy for a Gaussian random variable $X : \mathcal{N}(0, \sigma^2)$:

Let $X \sim \mathcal{N}(0, \sigma^2)$. The probability density function (pdf) of X is:

$$f_X(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x^2}{2\sigma^2}\right).$$

The differential entropy is defined as:

$$h(X) = -\int_{-\infty}^{\infty} f_X(x) \log f_X(x) \, dx.$$

Substitute the expression for $f_X(x)$:

$$h(X) = -\int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x^2}{2\sigma^2}\right) \log\left(\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x^2}{2\sigma^2}\right)\right) dx.$$

Simplify the logarithm:

$$\log f_X(x) = \log\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right) - \frac{x^2}{2\sigma^2}.$$

Now compute:

$$h(X) = -\int_{-\infty}^{\infty} f_X(x)\left[\log\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right) - \frac{x^2}{2\sigma^2}\right] dx.$$

Split the integral:

$$h(X) = -\log\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)\underbrace{\int f_X(x)dx}_{=1} + \frac{1}{2\sigma^2}\underbrace{\int x^2 f_X(x)dx}_{=\mathbb{E}[X^2]=\sigma^2}.$$

So:

$$h(X) = -\log\left(\frac{1}{\sqrt{2\pi\sigma^2}}\right) + \frac{1}{2} = \log\left(\sqrt{2\pi\sigma^2}\right) + \frac{1}{2}.$$

Final Result:

$$h(X) = \frac{1}{2}\log\big(2\pi e\sigma^2\big).$$

Note:

- The entropy is maximized for a given variance when the distribution is Gaussian.

(20250325#231)

Show that differential entropy $h(X+\mu) = h(X)$:

The differential entropy $h(X)$ of a continuous random variable X with probability density function (PDF) $f_X(x)$ is defined as:

$$h(X) = -\int_{-\infty}^{\infty} f_X(x)\log f_X(x)\,dx$$

where the logarithm is typically taken to be the natural logarithm (base e), in which case the entropy is measured in nats. If the base is 2, the entropy is measured in bits. The base of the logarithm does not affect the property we are about to show.

Proof of $h(X+\mu) = h(X)$

Let $Y = X + \mu$, where μ is a constant. We want to find the differential entropy of Y, denoted as $h(Y)$. First, we need to find the probability density function of Y, $f_Y(y)$.

The cumulative distribution function (CDF) of Y is given by:

$$F_Y(y) = P(Y \leq y) = P(X + \mu \leq y) = P(X \leq y - \mu) = F_X(y - \mu)$$

where $F_X(x)$ is the CDF of X.

The PDF is the derivative of the CDF with respect to its argument:

$$f_Y(y) = \frac{d}{dy} F_Y(y) = \frac{d}{dy} F_X(y - \mu)$$

Using the chain rule, we get:

$$f_Y(y) = f_X(y - \mu) \cdot \frac{d}{dy}(y - \mu) = f_X(y - \mu) \cdot 1 = f_X(y - \mu)$$

So, the PDF of $Y = X + \mu$ is $f_Y(y) = f_X(y - \mu)$.

Now, we can find the differential entropy of Y:

$$h(Y) = -\int_{-\infty}^{\infty} f_Y(y) \log f_Y(y) \, dy$$

Substitute $f_Y(y) = f_X(y - \mu)$ into the integral:

$$h(Y) = -\int_{-\infty}^{\infty} f_X(y - \mu) \log f_X(y - \mu) \, dy$$

Let's perform a change of variables. Let $x = y - \mu$, so $y = x + \mu$, and $dy = dx$. The limits of integration remain from $-\infty$ to ∞.

$$h(Y) = -\int_{-\infty}^{\infty} f_X(x) \log f_X(x) \, dx$$

We recognize this integral as the definition of the differential entropy of X, $h(X)$. Therefore,

$$h(Y) = h(X)$$

$$h(X + \mu) = h(X)$$

We have shown that the differential entropy of a continuous random variable is invariant under translation. Adding a constant to the random variable does not change its differential entropy. This is because the shape of the probability density function remains the same, only its location is shifted.

(20250325#232)

Show that differential entropy $h(cX) = h(X) + \log |c|$.

The differential entropy $h(X)$ of a continuous random variable X with probability density function (PDF) $f_X(x)$ is defined as:

$$h(X) = -\int_{-\infty}^{\infty} f_X(x) \log f_X(x)\, dx$$

where log denotes the natural logarithm.

Proof of $h(cX) = h(X) + \log |c|$

Let $Y = cX$, where c is a non-zero constant. We want to find the differential entropy of Y, denoted as $h(Y)$. First, we need to find the probability density function of Y, $f_Y(y)$.

The cumulative distribution function (CDF) of Y is given by: If $c > 0$:

$$F_Y(y) = P(Y \leq y) = P(cX \leq y) = P\left(X \leq \frac{y}{c}\right) = F_X\left(\frac{y}{c}\right)$$

If $c < 0$:

$$F_Y(y) = P(Y \leq y) = P(cX \leq y) = P\left(X \geq \frac{y}{c}\right) = 1 - P\left(X < \frac{y}{c}\right) = 1 - F_X\left(\frac{y}{c}\right)$$

Now, we find the PDF by differentiating the CDF with respect to y:

Case 1: $c > 0$

$$f_Y(y) = \frac{d}{dy}F_Y(y) = \frac{d}{dy}F_X\left(\frac{y}{c}\right) = f_X\left(\frac{y}{c}\right) \cdot \frac{1}{c} = \frac{1}{c}f_X\left(\frac{y}{c}\right)$$

Case 2: $c < 0$

$$f_Y(y) = \frac{d}{dy}\left(1 - F_X\left(\frac{y}{c}\right)\right) = -f_X\left(\frac{y}{c}\right) \cdot \frac{1}{c} = -\frac{1}{c}f_X\left(\frac{y}{c}\right)$$

Combining both cases, we can write the PDF of Y as:

$$f_Y(y) = \frac{1}{|c|}f_X\left(\frac{y}{c}\right)$$

Now, we can find the differential entropy of Y:

$$h(Y) = -\int_{-\infty}^{\infty} f_Y(y) \log f_Y(y)\, dy$$

Substitute the expression for $f_Y(y)$:

$$h(Y) = -\int_{-\infty}^{\infty} \frac{1}{|c|}f_X\left(\frac{y}{c}\right) \log\left(\frac{1}{|c|}f_X\left(\frac{y}{c}\right)\right)\, dy$$

Using the property of logarithms $\log(ab) = \log a + \log b$:

$$h(Y) = -\int_{-\infty}^{\infty} \frac{1}{|c|} f_X\left(\frac{y}{c}\right) \left(\log\left(\frac{1}{|c|}\right) + \log f_X\left(\frac{y}{c}\right)\right) dy$$

$$h(Y) = -\int_{-\infty}^{\infty} \frac{1}{|c|} f_X\left(\frac{y}{c}\right) (-\log|c|)\, dy - \int_{-\infty}^{\infty} \frac{1}{|c|} f_X\left(\frac{y}{c}\right) \log f_X\left(\frac{y}{c}\right)\, dy$$

$$h(Y) = \log|c| \int_{-\infty}^{\infty} \frac{1}{|c|} f_X\left(\frac{y}{c}\right)\, dy - \int_{-\infty}^{\infty} \frac{1}{|c|} f_X\left(\frac{y}{c}\right) \log f_X\left(\frac{y}{c}\right)\, dy$$

Let's perform a change of variables in both integrals. Let $x = \frac{y}{c}$, so $y = cx$, and $dy = |c|\, dx$. The limits of integration remain from $-\infty$ to ∞.

For the first integral:

$$\int_{-\infty}^{\infty} \frac{1}{|c|} f_X\left(\frac{y}{c}\right)\, dy = \int_{-\infty}^{\infty} \frac{1}{|c|} f_X(x)|c|\, dx = \int_{-\infty}^{\infty} f_X(x)\, dx = 1$$

For the second integral:

$$\int_{-\infty}^{\infty} \frac{1}{|c|} f_X\left(\frac{y}{c}\right) \log f_X\left(\frac{y}{c}\right)\, dy = \int_{-\infty}^{\infty} \frac{1}{|c|} f_X(x) \log f_X(x)|c|\, dx = \int_{-\infty}^{\infty} f_X(x) \log f_X(x)\, dx = -h(X)$$

Substitute these results back into the expression for $h(Y)$:

$$h(Y) = \log|c| \cdot 1 - (-h(X))$$

$$h(Y) = \log|c| + h(X)$$

Thus, we have shown that:

$$h(cX) = h(X) + \log|c|$$

(20250325#233)

What is the relationship between entropy and differential entropy?

We explore how differential entropy $h(X)$ relates to the entropy of a quantized version of a continuous random variable X.

Setup:

- Let X be a continuous random variable with density $f_X(x)$, supported on a finite interval $S = [a, a + |S|]$, i.e., $F_X(x) = 0$ for $x \notin S$.

- Quantize the interval S by dividing it into bins of width Δ. The number of bins is $\frac{|S|}{\Delta}$.
- Define the quantized version of X as $X^\Delta \in \{x_0, x_1, \ldots, x_{k-1}\}$, where each bin corresponds to an interval:

$$x_i \in [a + i\Delta, a + (i+1)\Delta), \quad i = 0, 1, \ldots, \frac{|S|}{\Delta} - 1.$$

Discrete Entropy of Quantized Variable:

Let $p_i = \mathbb{P}(X \in [a + i\Delta, a + (i+1)\Delta)) \approx f_X(x_i)\Delta$, for some representative point $x_i \in [a + i\Delta, a + (i+1)\Delta)$. Then:

$$H(X^\Delta) = -\sum_i p_i \log p_i \approx -\sum_i f_X(x_i)\Delta \log(f_X(x_i)\Delta).$$

Separate the log term:

$$H(X^\Delta) \approx -\sum_i f_X(x_i)\Delta \log f_X(x_i) - \sum_i f_X(x_i)\Delta \log \Delta.$$

Factor and approximate as an integral:

$$H(X^\Delta) \approx -\int_a^{a+|S|} f_X(x) \log f_X(x)\, dx + \log\left(\frac{1}{\Delta}\right) \int_a^{a+|S|} f_X(x)\, dx.$$

Since the integral of the density over the support is 1:

$$H(X^\Delta) \approx h(X) + \log\left(\frac{1}{\Delta}\right).$$

Conclusion:

$$\boxed{H(X^\Delta) = h(X) + \log\left(\frac{1}{\Delta}\right)}$$

Interpretation:

- As $\Delta \to 0$, the number of bins increases and the entropy $H(X^\Delta) \to \infty$.
- The differential entropy $h(X)$ captures the *rate offset* at which this divergence happens.
- Let $\Delta = 2^{-n}$. Then:
$$H(X^\Delta) = n + h(X).$$
- Hence, $h(X)$ represents the shift from the ideal bit rate for quantization at resolution Δ.

(20250325#234)

What is the physical intuition behind thinking of $h(X)$ as a rate offset at which the divergence of $H(X^\Delta) \to \infty$?

To understand the offset in the relationship

$$H(X^\Delta) = h(X) + \log\left(\frac{1}{\Delta}\right),$$

we explore the process of quantizing a continuous-valued random variable and how this affects the entropy.

- **Quantization as Binning:** When we approximate a continuous random variable X using bins of width Δ, we obtain a discretized version X^Δ. The number of bins is approximately $|S|/\Delta$, where S is the finite support of X.
- **Two Sources of Entropy:** The entropy $H(X^\Delta)$ consists of:
  - $\log(1/\Delta)$: the resolution term — the number of bits needed to specify which bin X falls into.
  - $h(X)$: the differential entropy — measures the spread or uncertainty in the density $f_X(x)$.
- **Offset as Information Density:** The term $h(X)$ is independent of the resolution Δ. It captures the intrinsic randomness of the distribution:
  - A uniform distribution has higher $h(X)$ due to evenly spread probability.
  - A concentrated distribution has lower $h(X)$.
- **Rate Interpretation:** For $\Delta = 2^{-n}$, we get:

  $$H(X^\Delta) = n + h(X)$$

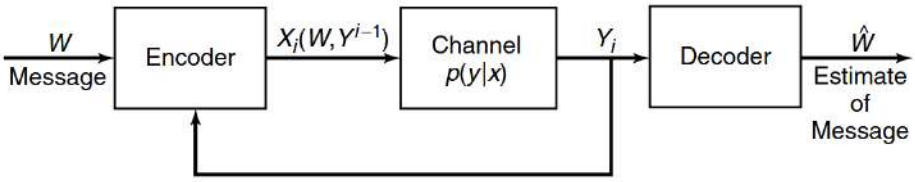
  implying that $h(X)$ is an offset in the number of bits needed as resolution increases.
- **Asymptotic Growth of Entropy:** As $n \to \infty$, $H(X^\Delta) \to \infty$, but the rate of this growth is linear, and $h(X)$ acts as a shift:

  $$h(X) = \lim_{n\to\infty}\left(H(X^\Delta) - \log\left(\frac{1}{\Delta}\right)\right)$$
- **Summary:** Differential entropy $h(X)$ reflects the *baseline uncertainty* per unit length of the real line. It determines how fast the entropy $H(X^\Delta)$ increases as we make the quantization finer, and provides a measure of the compressibility of a continuous source at high resolution.

(20250311#235)

Give an example for channels with feedback

Transmissions with acknowledgements



(20250311#236)

State the encoder function and how the input in each time index is going to look like for a channel with feedback:

Encoder:

$$f_i : \mathbb{W} \times B^{i-1} \to A$$

Transmissions will look like:

$$\begin{aligned}
X_1 &= f_1(w) \\
X_2 &= f_2(w, Y_1) \\
X_3 &= f_3(w, Y_1, Y_2) = f_3(W, Y^2) \\
X_4 &= f_4(w, Y_1, Y_2, Y_3) = f_4(w, Y^3)
\end{aligned}$$

and so on.

(20250311#237)

What is the relation between channel capacity with feedback and without feedback?

Achievable rate with feedback:

$$C_F = \sup\{R : R \text{ is achievable with feedback}\}$$

Clearly $C_F > C$ as we have additional information at each time step in the case of channel with feedback which we can choose to ignore or make use of inorder to optmize the rate of information transmission.

(20250311#238)

State and prove the theorem for the channel capacity of a discrete memoryless channel with and without feedback.

Theorem: For a discrete memoryless channel $(A, B, P_{Y|X})$,

$$C_F = C = \max_{P_X} I(X;Y)$$

Proof is as follows: Consider DMC without feedback:

$$\begin{aligned} I(W;\hat{W}) &\leq I(X^n;Y^n) \text{ using data processing inequality} \\ &\leq \sum_{i=1}^{n} I(X_i;Y_i) \text{given } X_i, Y_i \text{ is independent of other r.v.s)} \\ &\leq nC \end{aligned}$$

For a system with feedback,

$$\begin{aligned} I(W;\hat{W}) &\leq I(W;Y^n) \\ &\left\{= \sum_{i=1}^{n} I(W;Y_i \mid Y^{i-1}\right\} \\ &= H(Y^n) - H(Y^n|W) \\ &= H(Y^n) - \sum_{i=1}^{n} H(Y_i \mid Y^{i-1}, W) \end{aligned}$$

But we know, $X_i = f(W, Y^{i-1})$. Using this, we have

$$\begin{aligned} &= H(Y^n) - \sum_{i=1}^{n} H(Y_i \mid Y^{i-1}, W, X_i) \\ &= H(Y^n) - \sum_{i=1}^{n} H(Y_i \mid X_i) \\ &\leq \sum_{i=1}^{n} H(Y_i) - \sum_{i=1}^{n} H(Y_i \mid X_i) \\ &= \sum_{i=1}^{n} I(X_i;Y_i) \\ &= nC \end{aligned}$$

(20250311#239)

State source-channel separation priniciple:

We know these two things hold:

- Data compression: R > H
- Data transmission: R < C

The source comrepession tries to reduce the number of bits used to represent the source string, while the channel adds redundancy to the message for reliable communication → tries to increase the number of bits $\implies$ two opposite, competing intentions.

We ask the question: Whether $H < C$ is a necessary and sufficient condition for sending a source over a channel?

We have this two stage method: (Eg: sending digital music)

- Step-1: Compress the music to its most efficient form
- Step-2: Map the sequence of music code into channel codes

In the two stage method, data compression doesn't depend on channel and channel coding doesn't depend on source distribution.

Source-channel coding theorem states the following: If $V_1, V_2, \ldots, V_n$ satisfies AEP and $H(\mathcal{V}) \leq C$, there exists a source-channel code with $p(\hat{V} \neq V) \to 0$. Conversely for stationary processes, if $H(\mathcal{V}) > C$, probability of error is bounded away from 0.

(20250311#240)

State the implications of source-channel separation:

- Keeps design of source and channel coding separate
- Greatly simplifies the communication system design
- Source coding: find the most efficient representation of the source (removes redundancy)
- Channel coding: encodes the message to combat the noise and errors (introduces designed redundancy)

(20250311#241)

When is source-channel separation non-ideal?

The source-channel coding theorem assumes $n \to \infty$, point-to-point DMC.

Source-channel coding shouldn't be separated in situations where we have

- multiuser channels
- redundancy in the source is suited to the channel
- in speech and video transmissions, joint source-channel coding is valuable (since early 90s)

(20250311#242)

Why do we require $H(\mathcal{V}) \leq C$ in source-coding theorem? What happens otherwise?

(20250311#243)

Definition of the source-channel code:

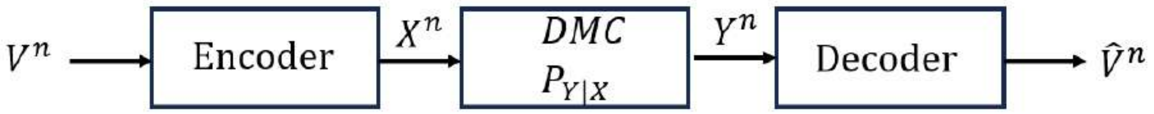
A **source-channel code** consists of an encoder and a decoder that together enable reliable communication of source data over a noisy channel, without explicitly separating compression and error correction.

- Let V be the source alphabet and V^n denote the space of source sequences of length n.
- Let A be the channel input alphabet, and B be the channel output alphabet.
- The encoder is a function:
$$f_n : V^n \to A^n$$
mapping the source sequence to the channel input sequence.
- The decoder is a function:
$$\phi_n : B^n \to V^n$$
mapping the received channel output to a reconstruction of the source sequence.

The channel is typically modeled as a conditional distribution $P_{B^n|A^n}$, representing the noisy transformation from A^n to B^n.

Objective: Design (f_n, ϕ_n) such that the probability of decoding error:

$$\mathbb{P}\left(\phi_n(Y^n) \neq V^n\right) \leq \epsilon$$



is small for a given $\epsilon > 0$, while keeping the encoding efficient in terms of rate and/or distortion.

Remark: In this joint approach, the source and channel coding are not treated independently. This is particularly useful when the separation principle is suboptimal due to complexity, delay, or non-asymptotic considerations.

(20250311#244)

When is a source transmissible across a channel?

Let $\{V_i\}_{i=1}^{\infty}$ be a source with source alphabet $\mathcal{V}$, and let the channel be a discrete memoryless channel (DMC) with input alphabet $\mathcal{A}$, output alphabet $\mathcal{B}$, and transition probability $P_{Y|X}$.

Definition: A source is said to be *transmissible across a channel* if $\forall \epsilon > 0$, there exists (n, f_n, ϕ_n), a sequence of encoder functions $f_n : \mathcal{V}^n \to \mathcal{A}^n$ and decoder functions $\phi_n : \mathcal{B}^n \to \mathcal{V}^n$ such that the probability of decoding error

$$\mathbb{P}\left(\phi_n(Y^n) \neq V^n\right) \leq \epsilon,$$

where $Y^n \sim P_{Y^n|X^n}(f_n(X^n))$.

Criterion: If the source is memoryless with entropy rate $H(P_X)$ and the channel has capacity C, then a sufficient condition for transmissibility is:

$$H(P_X) < C.$$

(20250311#245)

State the transmissibility theorem for a stationary and ergodic source over a DMC:

Let $\{X_i\}_{i=1}^{\infty}$ be a stationary and ergodic source over finite alphabet $\mathcal{V}$, with entropy rate

$$H = \lim_{n\to\infty} \frac{1}{n} H(X_1, X_2, \ldots, X_n).$$

Let the communication channel be a discrete memoryless channel (DMC) with input alphabet $\mathcal{A}$, output alphabet $\mathcal{B}$, and transition probability $P_{Y|X}$, with Shannon capacity C.

Theorem (Shannon, 1948): If $H < C$, then the source is *transmissible* across the channel, i.e., there exists a sequence of encoder-decoder pairs

$$f_n : \mathcal{V}^n \to \mathcal{A}^n, \quad \phi_n : \mathcal{B}^n \to \mathcal{V}^n$$

such that the probability of decoding error satisfies

$$\mathbb{P}\left(\phi_n(Y^n) \neq V^n\right) \to 0 \quad \text{as } n \to \infty,$$

where $Y^n \sim P_{Y^n|X^n}(f_n(X^n))$.

Converse: If the source is transmissible across the channel with vanishing error probability, then it must be that $H \leq C$.

Remark: This result formalizes the principle of *source-channel separation*: compression (source coding) and error protection (channel coding) can be designed independently and still achieve optimal performance, provided $H < C$. If $H > C$, the source is not transmissible over the DMC. For $H < C$, there is a goal for the source codes, there is a goal for the channel codes and there is division of labor.

(20250311#246)

Prove transmissibility theorem for stationary and ergodic source over a DMC:

Let $\{Y_n\}_{n\geq 1}$ be a stationary and ergodic source with entropy rate H. Let $(\mathcal{A}, \mathcal{B}, P_{Y|X})$ be a discrete memoryless channel (DMC) with capacity C.

- If $H < C$, then the source is **transmissible** over the DMC.
- If $H > C$, then the source is **not transmissible** over the DMC.

Remark: Separation Principle

There exists a conceptual and practical separation between:

- **Source coding**: Efficiently representing the source.
- **Channel coding**: Reliably transmitting information over a noisy channel.

This allows a division of labor—source compression followed by channel error correction.

Proof

(a) Achievability: If $H < C$

Define $\eta = \frac{C-H}{2}$.

We then have:

$$H < H + \eta = \frac{C+H}{2} < C$$

Source coding: By the source coding theorem for stationary ergodic sources, there exists a sequence of (n, r) block source codes such that:

- $\frac{r}{n} \leq H + \eta$
- Number of codewords: $2^{n(H+\eta)}$
- Source encoder error probability: $P_e^{(n,s)} \leq \varepsilon/2$

Channel coding: By the channel coding theorem, since $H + \eta < C$, there exists a sequence of (n, M_n) channel codes such that:

- $\frac{1}{n} \log M_n \geq C - \eta$
- Max probability of error: $P_{e,\max}^{(n,c)} \leq \varepsilon/2$

The number of compressed source codewords fits into the channel codebook:

$$2^{n(H+\eta)} < 2^{n(C-\eta)} = M_n$$

Hence, the entire pipeline (source compression → channel encoding → channel decoding → source reconstruction) succeeds with total error:

$$P_e^{(n)} \leq P_e^{(n,s)} + P_e^{(n,c)} \leq \frac{\varepsilon}{2} + \frac{\varepsilon}{2} = \varepsilon$$

Thus, the source is transmissible when $H < C$.

(b) Converse: If $H > C$

Assume a scheme exists that transmits the source reliably. Then Fano's inequality gives:

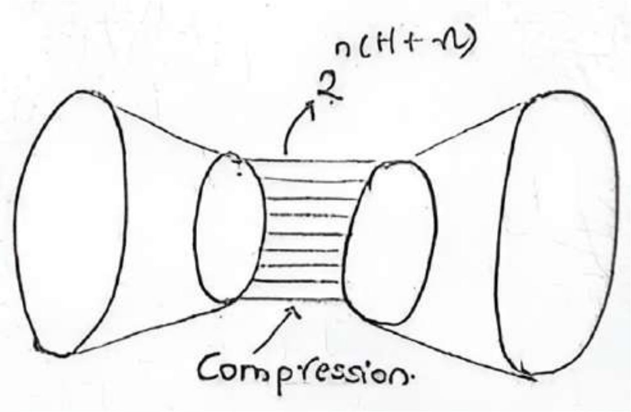
$$H(V^n \mid \hat{V}^n) \leq 1 + P_e^{(n)} \log |\mathcal{V}^n| = 1 + P_e^{(n)} n \log |\mathcal{V}|$$

Now,

$$H(V^n) = I(V^n; \hat{V}^n) + H(V^n \mid \hat{V}^n) \leq I(V^n; Y^n) + 1 + P_e^{(n)} n \log |\mathcal{V}| \leq nC + 1 + P_e^{(n)} n \log |\mathcal{V}|$$

Hence,

$$P_e^{(n)} \geq \frac{H(V^n) - nC - 1}{n \log |\mathcal{V}|}$$

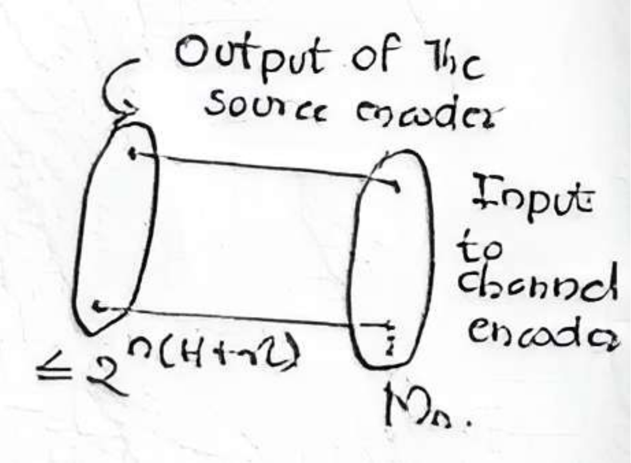


For stationary ergodic sources, $H(V^n) = nH + o(n)$, so:

$$P_e^{(n)} \geq \frac{nH - nC - 1}{n \log |\mathcal{V}|} = \frac{H - C - \frac{1}{n}}{\log |\mathcal{V}|}$$

Now, choose n such that $\frac{1}{n} \leq \frac{H-C}{2}$. Then:

$$P_e^{(n)} \geq \frac{H - C - \frac{1}{2}(H - C)}{\log |\mathcal{V}|} = \frac{H - C}{2 \log |\mathcal{V}|}$$

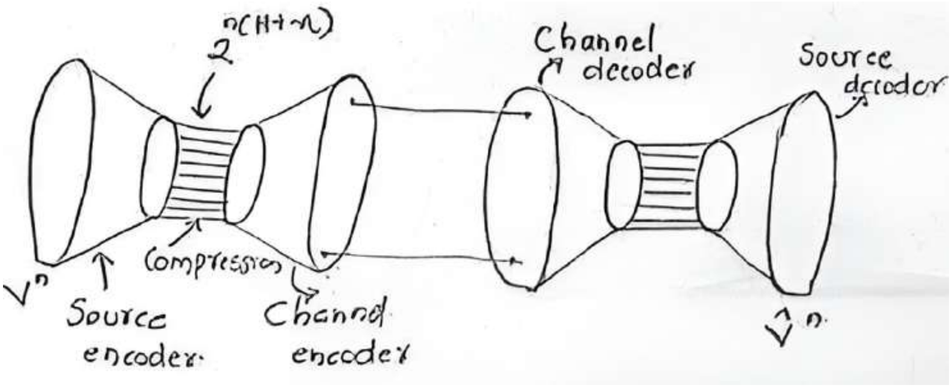


This shows that when $H > C$, the probability of error is bounded away from zero, so the source cannot be reliably transmitted.

Conclusion:

$$\boxed{\text{If } H < C \Rightarrow \text{ reliable transmission is possible.If } H > C \Rightarrow \text{ reliable transmission is impossible.}}$$

(20250311#247)



Prove that the capacity of a channel is not going to change even if we were to use maximum probability of error across all codewords in a codebook rather than average probability of error:

Theorem: The capacity of a discrete memoryless channel (DMC) remains the same whether we define it using the **maximum** probability of error or the **average** probability of error over the codebook.

Proof:

Let a DMC have capacity C. Consider an (n, M_n) code with encoder f_n and decoder ϕ_n, where:

- The average probability of error is
$$P_e^{(avg)} = \frac{1}{M_n} \sum_{i=1}^{M_n} \mathbb{P}(\phi_n(Y^n) \neq i \mid X^n = f_n(i)).$$
- The maximum probability of error is
$$P_e^{(max)} = \max_{i \in [M_n]} \mathbb{P}(\phi_n(Y^n) \neq i \mid X^n = f_n(i)).$$

It is clear that:
$$P_e^{(avg)} \leq P_e^{(max)}.$$

So any code that achieves small maximum error also achieves small average error. Thus:
$$C_{\max} \leq C_{\text{avg}} = C.$$

We now show the reverse inequality. That is, we can convert a good average-error code into a good maximum-error code without significantly reducing the rate.

Expurgation Argument:

Let there be an (n, M_n) code with average error probability $P_e^{(avg)} \leq \epsilon$. Let the error probability for codeword i be e_i. Then:
$$\frac{1}{M_n} \sum_{i=1}^{M_n} e_i \leq \epsilon.$$

Let us expurgate all codewords with error probability more than $\sqrt{\epsilon}$. The number of such codewords is at most:

$$\frac{1}{\sqrt{\epsilon}} \cdot \sum_{i=1}^{M_n} e_i \leq \frac{M_n \epsilon}{\sqrt{\epsilon}} = M_n \sqrt{\epsilon}.$$

So we can remove at most $M_n\sqrt{\epsilon}$ codewords and retain at least $M_n(1-\sqrt{\epsilon})$ codewords with individual error at most $\sqrt{\epsilon}$. Let the new code have $M_n' = \lfloor M_n(1-\sqrt{\epsilon}) \rfloor$ codewords.

Then:

$$\frac{1}{n}\log M_n' \geq \frac{1}{n}\log M_n + \frac{1}{n}\log\bigl(1-\sqrt{\epsilon}\bigr).$$

Taking limits:

$$\lim_{n\to\infty} \frac{1}{n}\log M_n' = \lim_{n\to\infty} \frac{1}{n}\log M_n.$$

Thus, any rate achievable with small average error is also achievable with small maximum error. Hence:

$$C_{\text{max}} \geq C_{\text{avg}}.$$

Therefore:

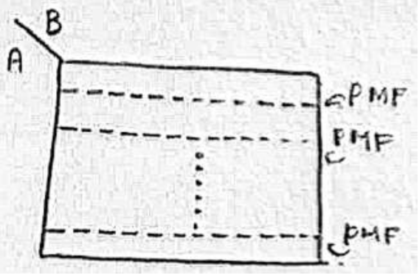
$$C_{\text{max}} = C_{\text{avg}} = C.$$

□

(20250304#248)

Define discrete channel:

A discrete channel is a tuple $(A, B, P_{Y|X})$ where $|A| < \infty$, $|B| < \infty$ and $P_{Y|X}$ is a stochastic matrix (each row of that matrix corresponds to $a_1, a_2, \ldots, a_{|A|}$ and columns corresponds to $b_1, b_2, \ldots, b_{|B|}$).



(20250304#249)

What is a discrete memoryless channel?

A discrete memoryless channel (DMC) also denoted $(A, B, P_{Y|X})$ is a sequence (meaning we're using the channel n times) of discrete channels where

$$P_{Y^n|X^n}(b^n|a^n)_{n\geq 1} = \prod_{i=1}^{n} P_{Y|X}(b_i|a_i)$$

Here the transition matrix is the same for each input, but what is input into the channel can be different.

(20250304#250)

Come up with a scheme for noisy typewriter example where we can pack 2 or more bits per transition.

(20250304#251)

Give the formal definition of a discrete memoryless channel:

An (n, M_n) code of a discrete memoryless channel $(A, B, P_{Y|X})$ is made of the following:

1. A message set, $\mathbb{W}_n = \{1, 2, 3, \ldots, M_n\}$
2. An encoder $f_n : \mathbb{W}_n \to A^n$, which maps each message to an n-letter input string.
3. A code: $c_n = \{f_n(1), f_n(2), f_n(3), \ldots, f_n(M_n) \subseteq A^n$, i.e., this code just ends up being a subset of the alphabet set A^n.
4. A decoder: $\phi_n : B^n \to \mathbb{W}^n$, where the channel output is stochastic.

The output of the channel for a chosen message w will be governed by the transition probability matrix:

$$P_{Y^n|X^n}\left(b^n|f_n(w)\right) = \prod_{i=1}^{n} P_{Y|X}\left(b_i|(f_n(w))_i\right)$$

(20250304#252)

What are the performance parameters for DMCs?

An (n, M_n) code has the following properties:

1. Rate measured in number of bits transmitted per channel use.
$$\text{Rate} = \frac{\log M_n}{n} \text{ bits per channel use}$$
2. $P_{e,w}^{n,(C_n)} \to$ probability of error given a message w is transmitted:
$$P_{e,w}^{n,(C_n)} = P_{Y^n|X^n}\left(\phi_n(Y^n) \neq w|f_n(w)\right)$$
We can have average probability across all the messages as well:
$$P_e^{n,(C_n)} \to \text{ average probability of error across all the messages}$$
We can either use average probability of error or maximum probability of error across all the messages. It won't affect the final capacity of the channel as we keep increasing $n \to \infty$.
$$P_e^{(n)}(c_n) = \frac{1}{M_n}\sum_{i=1}^{M_n} P_{e,w}^n$$
assuming uniform probability distribution over the message set.
$$P_e^{(n)}(c_n) = \max_{1\leq w\leq M_n} P_{e,w}^n$$
another way of defining probability of error. This takes the maximum across all possible errors. This doesn't need to have uniform probability distribution.

(20250304#253)

It can be shown that even if we were to use average probability of error or maximum probability of error, we're going to end up with the same capacity. So doesn't that mean the capacity is agnostic to the distribution used?

(20250304#254)

When do we say that a rate R is achievable?

A rate R is achievvable if for every $0 < \epsilon < 1$, and every $\eta > 0$, $\exists$ sequence of (n, M_n) codes (indexed by $n = 1, 2, 3, \ldots$) such that the probabillity of error $P_e^n(c_n) \leq \epsilon$ for all sufficnetly large n, and

$$\frac{\log M_n}{n} \geq R - \eta$$

That is, for first few n, our rate may not be $\leq \epsilon$, but eventually it will be true no matter the value of ϵ or η.

In layman terms: achievability of rates means that we can reliably send information over a noisy channel at a certain speed (rate) with an error that becomes negligible as messages get longer. In simple terms, it tells us how fast we can communicate without losing information, even when there's noise.

(20250304#255)

Give some remarks on the achievability of rates for a DMC:

---

1. Rate 0 is achievable. This is attained by taking $M_n = 1$ for all n

   $$\frac{\log M_n}{n} = 0$$

   It is possible to get exponentially fast decay of error with $n \to \infty$.

2. If rate R is achievable, then so is any $R' \in [0, R]$, i.e., if R is achievable, then any smaller rate is also possible. In terms of rate, one may define capacity of the DMC $(A, B, P_{Y|X})$ as

   $$C = \sup\{R : R \text{ is achievable}\}$$

3. The set of achievable rates is a closed set.

   If $R_n \to_{n\to\infty} R$, R_ns are achievable, then R is achievable.

Question: Is $C \geq 0$?

Yes, the channel capacity C is always non-negative.

To understand this intuitively, consider that the number of distinct messages M_n we can send over a noisy channel grows exponentially with block length n, provided we are operating below capacity.

For example, suppose we choose a communication rate $R = 0.5$, and allow a small gap $\eta = 0.1$. Then, we require that:

$$\frac{\log M_n}{n} \geq R - \eta = 0.4$$

This implies:

$$M_n \geq 2^{0.4n}$$

So, the number of messages we can send grows exponentially with n, confirming that the rate is positive and hence $C \geq 0$.

(20250304#256)

Prove this statement: If $R_n \to_{n\to\infty} R$, R_ns are achievable, then R is achievable.

Given: A sequence of rates $\{R_n\}$ such that $R_n \to R$ as $n \to \infty$, and each R_n is achievable. This means for every n, there exists a code of blocklength n with rate R_n and error probability $\varepsilon_n \to 0$ as $n \to \infty$.

To show: R is achievable.

Proof: Fix any $\eta > 0$. Since $R_n \to R$, there exists N such that for all $n \geq N$, we have

$$|R_n - R| < \eta \quad \Rightarrow \quad R_n > R - \eta.$$

Let $n \geq N$ and consider the code achieving rate R_n with vanishing error probability $\varepsilon_n \to 0$. Since $R_n > R - \eta$, this code also achieves a rate of at least $R - \eta$, for arbitrarily small $\eta > 0$, with vanishing error.

Therefore, for any $\eta > 0$, there exists a sequence of codes of increasing blocklength n such that:

$$\frac{\log M_n}{n} \geq R - \eta, \quad \text{with } \varepsilon_n \to 0.$$

This means $R - \eta$ is achievable for all $\eta > 0$. By the definition of achievable rates, this implies that R is achievable.

□

(20250304#257)

Capacity formula definition for a DMC

The capacity of the DMC $(A, B, P_{Y|X})$ is

$$C = \max_{P_X} I(X;Y)$$

(20250304#258)

Give some remarks based on the formula for channel capacity of a discrete memoryless channel:

---

1. $0 < \epsilon < 1 \rightarrow$ We don't ask for each specific ϵ. Independence of ϵ ($\epsilon > 0$). This means that $C(\epsilon)$ based on $R(\epsilon)$ goes to C no matter what, as $n \rightarrow \infty$.

2. Single letter characterization

3. C is concave in P_X Mutual information $I(X;Y)$ has two key properties:

   1. **Concavity in P_X for fixed $P_{Y|X}$**:

   $$I(\lambda P_X^{(1)} + (1-\lambda)P_X^{(2)}; Y) \geq \lambda I(P_X^{(1)}; Y) + (1-\lambda) I(P_X^{(2)}; Y)$$

   This holds because:

   - $I(X;Y) = H(Y) - H(Y|X)$
   - $H(Y)$ is *concave* in P_X (since entropy is concave)
   - $H(Y|X)$ is *linear* in P_X (it's an average of conditional entropies)
   - Concave − Linear = Concave

   2. **Maximum of Concave Functions is Concave**: The capacity C is the pointwise maximum of $I(X;Y)$ over all P_X:

   $$C(P_X) = \sup_{P_X} I(X;Y)$$

   The supremum of a family of concave functions remains concave.

Implications

- The concavity guarantees a **unique global maximum** (capacity-achieving distribution P_X^*)
- Enables efficient optimization (e.g., Blahut-Arimoto algorithm)
- Explains why capacity is well-defined for DMCs

$$\boxed{C \text{ is concave in } P_X \text{ because mutual information } I(X;Y) \text{ is concave in } P_X \text{ for fixed } P_{Y|X}.}$$

(20250304#259)

Will the probability simplex in the probability space corresponding to channel input alphabet A correspond to a convex hull?

Let the input alphabet be $A = \{x_1, x_2, \ldots, x_k\}$. The **probability simplex** over A is defined as:

$$\Delta^{k-1} = \left\{ (p_1, \ldots, p_k) \in \mathbb{R}^k \;\middle|\; p_i \geq 0,\ \sum_{i=1}^{k} p_i = 1 \right\}$$

This set represents all possible probability distributions over the finite alphabet A.

Now define δ_i as the point mass (Dirac distribution) at symbol x_i, i.e.,

$$\delta_i = (0, \ldots, 0, 1, 0, \ldots, 0) \in \mathbb{R}^k$$

with 1 in the i-th position.

Then the convex hull of these point masses is:

$$\operatorname{conv}(\{\delta_1, \ldots, \delta_k\}) = \left\{ \sum_{i=1}^{k} \lambda_i \delta_i \;\middle|\; \lambda_i \geq 0,\ \sum_{i=1}^{k} \lambda_i = 1 \right\}$$

This is exactly the probability simplex Δ^{k-1}, confirming that the simplex is the convex hull of the point masses over the alphabet A.

(20250304#260)

Why the maximum P_X in the definition of capacity always exists?

We consider the channel capacity defined as:

$$C = \max_{P_X} I(X;Y)$$

and aim to show that this maximum is always attained (i.e., the maximum exists) for a discrete memoryless channel with finite input and output alphabets.

1. Domain of Maximization is a Compact Set

The maximization is over all input distributions P_X on a finite input alphabet $A = \{x_1, x_2, \ldots, x_k\}$. The set of all such distributions forms the probability simplex:

$$\mathcal{P} = \left\{ (p_1, \ldots, p_k) \in \mathbb{R}^k \;\middle|\; p_i \geq 0, \; \sum_{i=1}^{k} p_i = 1 \right\}$$

This set $\mathcal{P}$ is closed and bounded in $\mathbb{R}^k$, and hence compact.

2. Mutual Information is Continuous in P_X

For a fixed channel transition probability $P_{Y|X}$, the mutual information $I(X;Y)$ is a continuous function of the input distribution P_X. In particular, it is given by:

$$I(X;Y) = H(Y) - H(Y|X)$$

where both entropy terms depend continuously on P_X. Therefore, $I(X;Y)$ is continuous on the simplex $\mathcal{P}$.

3. Extreme Value Theorem

By the **Extreme Value Theorem**:

> A continuous function on a compact set attains its maximum.

Since $I(X;Y)$ is continuous and $\mathcal{P}$ is compact, the maximum is achieved at some distribution $P_X^* \in \mathcal{P}$, i.e.,

$$C = I(X;Y)\big|_{P_X = P_X^*}$$

Conclusion

The maximum in the capacity expression exists because we are maximizing a continuous function (mutual information) over a compact set (the probability simplex). Therefore, the channel capacity C is always attained at some input distribution P_X^*.

(20250304#261)

Give the proof for Shannon's (1948) channel coding theorem:

Let $(A, B, P_{Y|X})$ be a discrete memoryless channel (DMC), where A is the input alphabet, B is the output alphabet, and $P_{Y|X}$ is the transition probability. Let C denote the channel capacity defined by:

$$C = \max_{P_X} I(X;Y),$$

where $I(X;Y)$ is the mutual information between input X and output Y.

The theorem has two parts:

1. **Achievability:** For any rate $R < C$, there exists a sequence of codes such that the probability of decoding error tends to zero as $n \to \infty$.
2. **Converse:** For any sequence of codes with vanishing error probability, the rate R must satisfy $R \leq C$.

Achievability Proof (Random Coding Argument)

Let $R < C$, and fix $\delta > 0$. Choose an input distribution P_X such that $I(X;Y) - 4\delta > R$. Define n-length codewords as follows:

- Generate $M_n = \lfloor 2^{nR} \rfloor$ codewords $x^n(1), \dots, x^n(M_n)$, each drawn i.i.d. according to P_X.

$$\frac{\log\left(2^{nR} - 1\right)}{n} \leq \frac{\log M_n}{n} \leq \frac{\log 2^{nR}}{n} \to_{n\to\infty} R$$

- **Encoding:** Message w is encoded to codeword $x^n(w)$. Pick the codewords randomly!

$$c_n = \{x^n(1), x^n(2), \dots, x^n(M_n)\}$$

  where for each $w \in \{1, 2, \dots, M_n\}$,

$$x_n(w) = \{x_1(w), x_2(w), \dots, x_n(w)\}$$

  and $x_i(w) \sim P_x$ i.i.d. Realization of the codebook c_n is the random variable corresponding to a random matrix of dimension $n \times M_n$.

- **Decoding:** Given received sequence Y^n, find the unique w such that $(X^n(w), Y^n)$ are jointly typical.

  Using maximum likelihood (ML) estimator for the decoder is the best option. But for academic purposes we'll look at a suboptimal decoder based on typical sets because of its ease of analysis.

  Typical set decoder $A(n, \delta)$ "jointly typical" $\subset A^n \times B^n$ if $\exists$ unique $\hat{w}$ such that $(x^n(w), y^n) \in A(n, \delta)$.

  Note: We reveal the realized c_n to both the encoder and the decoder.

  Encoder: $w \to f_n(w)$, where each w gets mapped to a column on our codebook. The channel's role is to make it difficult for the analyst to recover back the input codeword. Codebook representation:

$$c_n = \begin{bmatrix} x_1(1) & x_1(2) & \cdots & x_1(n) \\ x_2(1) & x_2(2) & \cdots & x_2(n) \\ \vdots & \vdots & \ddots & \vdots \\ x_{M_n}(1) & x_{M_n}(2) & \cdots & x_{M_n}(n) \end{bmatrix}$$

  where each column in codebook makes up a codeword.

- Channel: $Y^n \sim P_{Y^n|X^n}(\cdot|f_n(w))$

Error Analysis: The average probability of error $P_e^{(n)}$ satisfies:

1. The probability that the correct codeword is not jointly typical with Y^n tends to 0 as $n \to \infty$.
2. The probability that any other codeword is jointly typical with Y^n is bounded by:

$$(M-1) \cdot 2^{-n(I(X;Y)-\delta)} \leq 2^{nR} \cdot 2^{-n(I(X;Y)-\delta)}.$$

   In short, error can occur if the correct codeword doesn't fall into the jointly typical set $(x^n(w), y^n) \notin A(n, \delta)$ or error can occur if $\exists$ another $\hat{w} \neq w$ which is $(x^n(w), y^n)$ is also typical (i.e., $\in A(n, \delta)$).

If $R < I(X;Y)$, then this total error probability $P_e^{(n)} \to 0$ as $n \to \infty$. Hence, any $R < C$ is achievable.

Converse Proof

Assume a sequence of codes (M_n, n) with $M_n = 2^{nR}$ and vanishing error probability. Let X^n be the input and Y^n be the output.

By Fano's inequality:

$$H(W|Y^n) \leq 1 + P_e^{(n)} \cdot nR,$$

where W is the message index. Then,

$$H(W) = I(W;Y^n) + H(W|Y^n) \leq I(W;Y^n) + n\varepsilon_n,$$

with $\varepsilon_n \to 0$ as $n \to \infty$.

Since $W \to X^n \to Y^n$ forms a Markov chain,

$$I(W; Y^n) \leq I(X^n; Y^n) = \sum_{i=1}^{n} I(X_i; Y_i) \leq nC.$$

Hence,

$$nR \leq nC + n\varepsilon_n \Rightarrow R \leq C + \varepsilon_n.$$

As $n \to \infty$, $\varepsilon_n \to 0$, so $R \leq C$.

Conclusion

Any rate $R < C$ is achievable (achievability), and no rate $R > C$ is achievable with vanishing error (converse). Therefore, C is the maximum reliable communication rate over a DMC.

(20250304#262)

Why is choosing a random codebook not a bad idea?

Why Random Codebooks Are Not a Bad Idea (Mathematical Justification): Let C_n denote a random codebook (with M_n codewords of length n), drawn according to some codebook distribution P_{C_n}. Let $P_{e,w}^{(n)}(c_n)$ denote the probability of decoding error when message w is sent using the deterministic codebook c_n.

We begin by analyzing the average probability of error over all random codebooks:

$$\sum_{c_n} P_{C_n}(c_n) \cdot \frac{1}{M_n} \sum_{w=1}^{M_n} P_{e,w}^{(n)}(c_n) = \mathbb{E}_{C_n} \left[\frac{1}{M_n} \sum_{w=1}^{M_n} P_{e,w}^{(n)}(C_n) \right].$$

where

$$P_{C_n}(c_n) = \prod_{i=1}^{n} \prod_{w=1}^{M_n} P_X\left((c_n)_{i,w}\right)$$

This is the **expected average error probability** when the codebook is randomly selected. The steps that follow are:

- Using properties of the channel and the random code construction, we can analyze this expectation using jointly typical decoding.

- For any rate $R < I(X; Y)$, the expectation can be shown to go to zero as $n \to \infty$ (using typicality arguments and the union bound).

- Therefore, since the expected error is small, there must exist at least one particular codebook $\hat{c}_n$ for which the average error satisfies:

$$\frac{1}{M_n}\sum_{w=1}^{M_n} P_{e,w}^{(n)}(\hat{c}_n) \le \mathbb{E}_{C_n}\left[\frac{1}{M_n}\sum_{w=1}^{M_n} P_{e,w}^{(n)}(C_n)\right].$$

  So the performance of a random codebook shows the existence of good deterministic codebooks.

Assume the codebook c_n is generated by drawing each codeword independently and identically according to a distribution P_X^n, and that the decoding rule is symmetric for all messages (e.g., based on typicality or ML decoding).

Since the channel is memoryless and all codewords are generated i.i.d., the error probability $P_{e,w}^{(n)}(c_n)$ is the same for all messages w. Thus,

$$P_{e,w}^{(n)}(c_n) = P_{e,1}^{(n)}(c_n) \quad \forall w.$$

Therefore, the average over all messages becomes:

$$\frac{1}{M_n}\sum_{w=1}^{M_n} P_{e,w}^{(n)}(c_n) = P_{e,1}^{(n)}(c_n).$$

So,

$$\sum_{c_n} P_{C_n}(c_n) \cdot \frac{1}{M_n}\sum_{w=1}^{M_n} P_{e,w}^{(n)}(c_n) = \mathbb{E}_{C_n}\left[P_{e,1}^{(n)}(c_n)\right].$$

Thus, choosing a random codebook is not only ”not a bad idea”, but also central to the proof of Shannon’s channel coding theorem.

(20250327#263)

Define multivariate differential entropy:

$X_1, X_2, \dots, X_n \sim P_{X_1,X_2,\dots X_n}$ with density $f_{X_1X_2\dots X_n}$ Then

$$h(X_1, X_2, \dots X_n) = \mathbb{E}\left[\log \frac{1}{f_{X_1,X_2,\dots,X_n}(X_1, X_2, \dots, X_n)}\right]$$

provided the expectation exists.

For a continuous random vector $\mathbf{X} \in \mathbb{R}^n$ with joint probability density function $f(\mathbf{x})$, the **differential entropy** is defined as:

$$h(\mathbf{X}) = -\int_{\mathcal{X}} f(\mathbf{x}) \log f(\mathbf{x})\, d\mathbf{x}$$

where $\mathcal{X}$ is the support of $\mathbf{X}$.

In particular,

$$h(X, Y) = \mathbb{E}\left[\log \frac{1}{f_{XY}(XY)}\right]$$

(20250327#264)

When is $h_{Y|X}$ defined? How is it defined?

The conditional differential entropy $h_{Y|X}$ is defined when:

1. **Joint Density Exists**:
   The random variables (X, Y) must have a well-defined joint probability density function $f_{X,Y}(x, y)$.

2. **Marginal Density Exists**:
   The marginal density $f_X(x) = \int f_{X,Y}(x, y)dy$ must exist for all x where $f_X(x) > 0$.

3. **Conditional Density Exists**:
   The conditional density $f_{Y|X}(y|x) = \frac{f_{X,Y}(x,y)}{f_X(x)}$ must be well-defined for all (x, y) where $f_X(x) > 0$.

When the above conditions are satisfied:

$$h_{Y|X} = -\iint f_{X,Y}(x, y) \log f_{Y|X}(y|x)\, dxdy$$

or

$$h(Y|X) = \mathbb{E}\left[\frac{1}{P_{Y|X}(Y|X)}\right]$$

and

$$h(X,Y) = h(X) + h(Y|X)$$

in which case both $f_X(X)$ and $f_{Y|X}(Y|X)$ needs to exist individually, or if both $h(X)$ and $h(Y|X)$ are $\rightarrow \infty$ or $-\infty$, then we've to check for the balance between those two terms such that $h(X,Y)$ is finite.

(20250327#265)

Give the expression for general chain rule for multivariate differential entropy:

$$\begin{aligned} h(X_1,\dots,X_n) &= h(X_1) + h(X_2|X_1) + \dots + h(X_n|X_1 \dots X_{n-1}) \\ &= h(X_1) + \sum_{i=2}^{n} h(X_i|X^{(i-1)}) \end{aligned}$$

(20250327#266)

Obtain a guess estimate for multivariate independent gaussian input's differential entropy:

Suppose

$$(X_1, X_2, \dots, X_n) = \mathcal{N}(\mu, \mathcal{K})$$

$$\begin{aligned} h(X_1,\dots,X_n) &= \sum_{i=1}^{n} \frac{1}{2}\log\big(2\pi\sigma_i^2\big) \\ &= \frac{n}{2}\log(2\pi e) + \frac{1}{2}\log(|\mathcal{K}|) \end{aligned}$$

Guess based on answer for independent gaussians.

(20250327#267)

Prove that the guess estimate is in fact the correct estimate of differential entropy of multivariate independent gaussians:

$$f_{X_1\dots X_n} = \frac{1}{(2\pi)^{n/2}|K|^{1/2}} e^{-\frac{1}{2}x^T K^{-1} x}$$

Without loss of generality, $\mu = 0$ (as $h(\mu + x) = h(x)$.

$$\log\left(\frac{1}{f_{X_1\dots X_n}}\right) = \frac{n}{2}\log(2\pi) + \frac{1}{2}\log(|K|) + (\frac{1}{2}x^T K^{-1} x)\log e$$

$$\begin{aligned}
\mathbb{E}\left[X^T K^{-1} X\right] &= \mathbb{E}\left[tr\left(X^T K^{-1} X\right)\right] \\
&= \mathbb{E}\left[tr\left(K^{-1} X^T X\right)\right] \\
&= tr\left(\mathbb{E}\left[K^{-1} X^T X\right]\right) \\
&= tr\left(K^{-1} K\right) \\
&= tr(I_d) \\
&= n
\end{aligned}$$

Plug this into $\log(\dots)$ expression,

$$\begin{aligned}
h(X_1, \dots, X_n) &= \mathbb{E}\left[\log\left(\frac{1}{f_{X_1\dots X_n}}\right)\right] \\
&= \mathbb{E}\left[\frac{n}{2}\log(2\pi)\right] + \mathbb{E}\left[\frac{1}{2}\log(|K|)\right] + \mathbb{E}\left[(\frac{1}{2}x^T K^{-1} x)\log e\right] \\
&= \frac{n}{2}\log(2\pi e) + \frac{1}{2}\log(|K|)
\end{aligned}$$

which is the same as our guess estimate.

(20250327#268)

Why will $tr(X^T K^{-1} X) = tr(K^{-1} X^T X)$?

(20250327#269)

Define relative entropy of continuous random variables:

Let P and Q be two distributions $\sim \mathbb{R}$. Then $D(P\|Q) =?$

$$D(P\|Q) \leq D([P]_{\mathcal{A}} + [Q]_{\mathcal{A}})$$

Using data prcoessing inequality, processing data can't increase the information content.

(20250327#270)

Obtain expression for relative entropy using partitioning approach for continuous case:

Let P and Q be two probability distributions over $\mathbb{R}$. The **relative entropy** or **Kullback–Leibler (KL) divergence** of P with respect to Q, denoted $D(P\|Q)$, is a measure of how one distribution diverges from another.

In the continuous case, if P and Q have densities f_P and f_Q, respectively, with respect to the Lebesgue measure, then:

$$D(P\|Q) = \int_{-\infty}^{\infty} f_P(x) \log\left(\frac{f_P(x)}{f_Q(x)}\right) dx = \mathbb{E}_P\left[\log\left(\frac{f_P(X)}{f_Q(X)}\right)\right]$$

From Discrete to Continuous: Partitioning Approach

To define $D(P\|Q)$ rigorously for arbitrary distributions on $\mathbb{R}$, we can approximate it using discrete relative entropy:

- Partition $\mathbb{R}$ into a finite number of measurable sets $\mathcal{A} = \{A_1, A_2, \ldots, A_k\}$.
- Induce discrete distributions on this partition:

$$[P]_{\mathcal{A}}(A_i) := P(A_i), \quad [Q]_{\mathcal{A}}(A_i) := Q(A_i)$$

- Then define the discrete relative entropy:

$$D([P]_{\mathcal{A}}\|[Q]_{\mathcal{A}}) = \sum_{i=1}^{k} P(A_i) \log\left(\frac{P(A_i)}{Q(A_i)}\right)$$

- Finally, define:

$$D(P\|Q) := \sup_{\mathcal{A}} D([P]_{\mathcal{A}}\|[Q]_{\mathcal{A}})$$

  where the supremum is over all finite measurable partitions $\mathcal{A}$ of $\mathbb{R}$.

Why the Supremum?

- Coarser partitions lose information: the uncertainty about whether the value lies on the left or right of a boundary introduces ambiguity.
- Finer partitions reveal more structure of the distributions.
- As $\mathcal{A}$ gets finer, the discrete divergence better approximates the true divergence.

- This is aligned with the **data processing inequality**: information content cannot increase under processing (e.g., under a coarser partitioning).

Alternative Expression with Radon–Nikodym Derivative

If $P \ll Q$ (i.e., P is absolutely continuous with respect to Q), and the Radon–Nikodym derivative $\frac{dP}{dQ}$ exists, then:

$$D(P\|Q) = \mathbb{E}_P\left[\log\left(\frac{dP}{dQ}(X)\right)\right]$$

Non-negativity

$$D(P\|Q) \geq 0$$

with equality if and only if $P = Q$ (almost everywhere). This can be proved using Jensen's inequality applied to the convex function $x \mapsto x \log x$, or by the fact that KL divergence measures the inefficiency of assuming Q when the true distribution is P.

Extension to Product Distributions

Suppose $P_{X^n} = P_{X_1} \times \cdots \times P_{X_n}$ and $Q_{X^n} = Q_{X_1} \times \cdots \times Q_{X_n}$, then:

$$D(P_{X^n}\|Q_{X^n}) = \sum_{i=1}^{n} D(P_{X_i}\|Q_{X_i})$$

if $X_1, \ldots, X_n$ are independent under both P and Q.

Summary

$$\boxed{\begin{aligned} D(P\|Q) &= \sup_{\mathcal{A}} D([P]_{\mathcal{A}}\|[Q]_{\mathcal{A}}) \\ &= \int f_P(x)\log\left(\frac{f_P(x)}{f_Q(x)}\right)dx \quad \text{(if densities exist)} \\ &= \mathbb{E}_P\left[\log\left(\frac{dP}{dQ}(X)\right)\right] \\ D(P\|Q) &\geq 0 \quad \text{with equality iff } P = Q \text{ a.e.} \end{aligned}}$$

(20250327#271)

Prove that differential entropy of Gaussian distribution is the largeest among all possible distributions:

Let $\mathbf{X} \in \mathbb{R}^n$ be a continuous random vector such that $\mathbb{E}[\mathbf{X}] = \mathbf{0}$ (mean zero) and $\mathbb{E}[\mathbf{X}\mathbf{X}^\top] = \mathbf{K}$ (covariance matrix $\mathbf{K}$). Let $\mathbf{X}_G \sim \mathcal{N}(0, \mathbf{K})$ be a multivariate Gaussian random vector with the same mean and covariance.

Then,

$$h(\mathbf{X}) \leq h(\mathbf{X}_G)$$

That is, among all random vectors with a fixed covariance matrix, the Gaussian distribution has the maximum differential entropy.

Remark: This result contrasts with the discrete case, where the uniform distribution maximizes the entropy.

Proof

Assume that $\mathbf{X}$ has a density function $f_\mathbf{X}$. Consider the Kullback–Leibler divergence between $\mathbf{X}$ and $\mathbf{X}_G$:

$$D(P_\mathbf{X} \| P_{\mathbf{X}_G}) = \mathbb{E}_\mathbf{X}\left[\log\left(\frac{f_\mathbf{X}(\mathbf{x})}{f_{\mathbf{X}_G}(\mathbf{x})}\right)\right]$$

Expanding the expectation:

$$D(P_\mathbf{X} \| P_{\mathbf{X}_G}) = -h(\mathbf{X}) + \mathbb{E}_\mathbf{X}\left[\log\left(\frac{1}{f_{\mathbf{X}_G}(\mathbf{x})}\right)\right]$$

Now, compute the expectation of the negative log Gaussian density. The density of $\mathbf{X}_G \sim \mathcal{N}(0, \mathbf{K})$ is:

$$f_{\mathbf{X}_G}(\mathbf{x}) = \frac{1}{(2\pi)^{n/2}|\mathbf{K}|^{1/2}} \exp\left(-\frac{1}{2}\mathbf{x}^\top \mathbf{K}^{-1}\mathbf{x}\right)$$

Then,

$$\begin{aligned}
\mathbb{E}_\mathbf{X}\left[\log\left(\frac{1}{f_{\mathbf{X}_G}(\mathbf{x})}\right)\right] &= \mathbb{E}_\mathbf{X}\left[\frac{n}{2}\log(2\pi) + \frac{1}{2}\log|\mathbf{K}| + \frac{1}{2}\mathbf{x}^\top\mathbf{K}^{-1}\mathbf{x} \cdot \log e\right] \\
&= \frac{n}{2}\log(2\pi) + \frac{1}{2}\log|\mathbf{K}| + \frac{1}{2}\log e \cdot \mathbb{E}\left[\mathbf{x}^\top\mathbf{K}^{-1}\mathbf{x}\right] \\
&= \frac{n}{2}\log(2\pi) + \frac{1}{2}\log|\mathbf{K}| + \frac{1}{2}\log e \cdot \mathrm{Tr}(\mathbf{K}^{-1}\mathbb{E}[\mathbf{X}\mathbf{X}^\top]) \\
&= \frac{n}{2}\log(2\pi) + \frac{1}{2}\log|\mathbf{K}| + \frac{1}{2}\log e \cdot \mathrm{Tr}(\mathbf{K}^{-1}\mathbf{K}) \\
&= \frac{n}{2}\log(2\pi) + \frac{1}{2}\log|\mathbf{K}| + \frac{n}{2}\log e \\
&= h(\mathbf{X}_G)
\end{aligned}$$

Therefore,

$$D(P_{\mathbf{X}} \| P_{\mathbf{X}_G}) = -h(\mathbf{X}) + h(\mathbf{X}_G)$$

But KL divergence is always non-negative:

$$D(P_{\mathbf{X}} \| P_{\mathbf{X}_G}) \geq 0 \quad \Rightarrow \quad h(\mathbf{X}) \leq h(\mathbf{X}_G)$$

Conclusion: The Gaussian distribution maximizes differential entropy among all distributions with a fixed mean and covariance.

(20250327#272)

Obtain expressions for mutual information and chain rule in continuous random variable context:

Let $X, Y \in \mathbb{R}$ be continuous random variables. Mutual information between X and Y, denoted by $I(X;Y)$, quantifies the reduction in uncertainty of one variable due to knowledge of the other.

Definition via Quantization:

For general (possibly continuous) random variables, we can define mutual information as:

$$I(X;Y) = \sup_{\mathcal{A},\mathcal{B}} I([X]_{\mathcal{A}}; [Y]_{\mathcal{B}})$$

where:

- $\mathcal{A}, \mathcal{B}$ are finite measurable partitions (quantizations) of $\mathbb{R}$,
- $[X]_{\mathcal{A}}$ and $[Y]_{\mathcal{B}}$ are discrete random variables obtained by mapping X and Y to the partition bins.

Note: the partitions $\mathcal{A}$ and $\mathcal{B}$ need not be the same.

Differential Entropy Formulation (When Densities Exist)

If the joint distribution $P_{X,Y}$ has a density $f_{X,Y}(x,y)$, and the marginals $f_X(x)$, $f_Y(y)$ exist, then:

$$I(X;Y) = \int_{\mathbb{R}^2} f_{X,Y}(x,y) \log\left(\frac{f_{X,Y}(x,y)}{f_X(x) f_Y(y)}\right) dx\, dy$$

Equivalently,

$$I(X;Y) = h(X) + h(Y) - h(X,Y)$$

This is a continuous analogue of the mutual information in the discrete case.

Conditional Entropy Formulation

If the conditional density $f_{Y|X}(y|x)$ exists, then:

$$I(X;Y) = \mathbb{E}\left[\log\left(\frac{f_{Y|X}(Y|X)}{f_Y(Y)}\right)\right] = h(Y) - h(Y|X)$$

This shows that mutual information measures the average reduction in uncertainty of Y due to knowledge of X.

Remarks

- In some situations, individual densities such as f_X or $f_{X,Y}$ may not exist, but we may still define f_Y as a mixture of conditional densities — e.g., via kernel smoothing or regularization.
- For instance, if $X = X + Z$, then f_Y may be interpreted as the convolution of a signal and noise density, which is a kind of mixture.
- **Caution**: Mixing discrete and differential entropies in the same expression is generally not well-defined due to differing units and interpretations (bits vs. nats, discrete vs. continuous).

Fundamental Properties

- $I(X;Y) \geq 0$, with equality if and only if $X \perp Y$ (i.e., X and Y are independent).
- From $I(X;Y) = h(Y) - h(Y|X)$, we immediately obtain:

  $$h(Y) \geq h(Y|X)$$

  Entropy decreases (or remains constant) upon conditioning.
- **Chain Rule for Differential Entropy:**
  Let $Y_1, Y_2, \ldots, Y_n$ be continuous random variables. Then:

  $$h(Y_1, Y_2, \ldots, Y_n) = \sum_{i=1}^{n} h(Y_i|Y^{i-1})$$

  where $Y^{i-1} = (Y_1, \ldots, Y_{i-1})$. In particular:

  $$h(Y_1, \ldots, Y_n) \leq \sum_{i=1}^{n} h(Y_i)$$

  with equality only if the variables are mutually independent.

(20250327#273)

Prove the converse of this theorem: The capacity of the gaussian channel wit hpower constraint P and noise σ^2 is

$$C = \frac{1}{2}\log\left(1+\frac{P}{\sigma^2}\right)$$

Theorem: The capacity of the additive white Gaussian noise (AWGN) channel with average power constraint P and noise variance σ^2 is:

$$C = \frac{1}{2}\log\left(1+\frac{P}{\sigma^2}\right) \quad \text{(in bits per channel use)}$$

Proof (Converse):

We consider the communication chain:

$$W \to X^n \to \text{Channel} \to Y^n \to \hat{W}$$

where:

- $W \in \{1, 2, \dots, M_n\}$ is the message,
- $X^n(W)$ is the codeword corresponding to message W,
- $Y^n = X^n + Z^n$ is the received vector, where $Z^n \sim \mathcal{N}(0, \sigma^2 I_n)$ is i.i.d. Gaussian noise,
- $\hat{W}$ is the decoder's estimate of W.

We wish to upper bound the mutual information between W and $\hat{W}$, ultimately obtaining:

$$I(W;\hat{W}) \leq \frac{n}{2}\log\left(1+\frac{P}{\sigma^2}\right)$$

Step 1: Data Processing Inequality:

$$I(W;\hat{W}) \leq I(X^n;Y^n)$$

Step 2: Channel is memoryless and additive Gaussian:

$$I(X^n;Y^n) = h(Y^n) - h(Y^n|X^n) = h(Y^n) - h(Z^n)$$

Since $Z^n \sim \mathcal{N}(0, \sigma^2 I_n)$, we know:

$$h(Z^n) = \frac{n}{2}\log(2\pi e\sigma^2)$$

Step 3: Upper bound for $h(Y^n)$:

By entropy maximization (Gaussian maximizes differential entropy for fixed variance), we have:

$$h(Y^n) \leq \sum_{i=1}^{n} h(Y_i)$$

Each $Y_i = X_i + Z_i$, with $Z_i \sim \mathcal{N}(0, \sigma^2)$, and X_i depending on the codeword.

Define the average input power at the i-th coordinate as:

$$P_i = \frac{1}{M_n} \sum_{w=1}^{M_n} X_i(w)^2$$

Then, the variance of Y_i is $\mathrm{Var}(Y_i) = P_i + \sigma^2$, and:

$$h(Y_i) \leq \frac{1}{2} \log\bigl(2\pi e(P_i + \sigma^2)\bigr)$$

Hence,

$$h(Y^n) \leq \sum_{i=1}^{n} \frac{1}{2} \log\bigl(2\pi e(P_i + \sigma^2)\bigr)$$

Step 4: Putting it together:

$$\begin{aligned}
I(X^n; Y^n) &= h(Y^n) - h(Z^n) \\
&\leq \sum_{i=1}^{n} \left[\frac{1}{2} \log\bigl(2\pi e(P_i + \sigma^2)\bigr)\right] - \frac{n}{2} \log\bigl(2\pi e \sigma^2\bigr) \\
&= \sum_{i=1}^{n} \frac{1}{2} \log\left(1 + \frac{P_i}{\sigma^2}\right)
\end{aligned}$$

Step 5: Using Jensen's Inequality:

The function $x \mapsto \log(1 + x)$ is concave, so:

$$\frac{1}{n} \sum_{i=1}^{n} \log\left(1 + \frac{P_i}{\sigma^2}\right) \leq \log\left(1 + \frac{1}{n\sigma^2} \sum_{i=1}^{n} P_i\right)$$

Define average input power:

$$\bar{P} = \frac{1}{n} \sum_{i=1}^{n} P_i \leq P$$

Thus,

$$I(W; \hat{W}) \leq I(X^n; Y^n) \leq \frac{n}{2} \log\left(1 + \frac{\bar{P}}{\sigma^2}\right) \leq \frac{n}{2} \log\left(1 + \frac{P}{\sigma^2}\right)$$

Conclusion:

This proves the *converse*, i.e., no communication scheme can achieve a rate higher than:

$$C = \frac{1}{2} \log \left(1 + \frac{P}{\sigma^2} \right)$$

bits per channel use, under the average power constraint P.

(20250401#274)

Explain the random coding used for the Gaussian channel with power constraint:

We're trying to prove the achievability for the Gaussian channel with power constraints:

We require to show that

$$C = \frac{1}{2}\log\left(1 + \frac{P}{\sigma^2}\right)$$

Consider a (n, M_n) random code defined as follows:

- The **codebook** consists of M_n codewords:

  $$\mathcal{C} = \{\mathbf{x}^n(1), \mathbf{x}^n(2), \ldots, \mathbf{x}^n(M_n)\}$$

  where each codeword is a vector:

  $$\mathbf{x}^n(w) = (x_1(w), x_2(w), \ldots, x_n(w)), \quad w = 1, 2, \ldots, M_n$$

- Each component $x_i(w)$ of every codeword is generated **independently and identically distributed (i.i.d.)** according to a Gaussian distribution:

  $$x_i(w) \sim \mathcal{N}(0, P'), \quad \text{where } P' < P$$

  Here:

  - $\mathcal{N}(0, P')$ denotes the Gaussian (normal) distribution with mean 0 and variance P'
  - P represents the power constraint of the channel
  - The choice $P' < P$ ensures the code satisfies the power constraint in expectation

(20250401#275)

Why choose $P' < P$?

For the case where $P' < P$, we proceed as follows:

- **Invoking the Law of Large Numbers (LLN)**:
  - Since we don't have direct control over the instantaneous P' values
  - We only require that $\mathbb{E}[P'] < P$ on average

  - By LLN, for large n:

$$\frac{1}{n}\sum_{i=1}^{n} x_i^2(w) \to P' \quad \text{almost surely}$$

- **Statistical Behavior**:
  - For any given codeword, approximately 50% of components may exceed P
  - However, the average power converges to $P' < P$
- **Restriction Requirement**:
  - Must constrain P' sufficiently below P to ensure:

$$\Pr\left(\frac{1}{n}\sum_{i=1}^{n} x_i^2(w) > P\right) \to 0 \quad \text{as } n \to \infty$$

  - Achieved by selecting $P' = P - \epsilon$ for some $\epsilon > 0$

Key Insight: The probabilistic construction satisfies the power constraint asymptotically while maintaining coding flexibility.

(20250401#276)

Prove the achievability part of the channel coding theorem for gaussian channel with average power constraint:'

Goal: Prove the achievability part of the capacity theorem for the Gaussian channel with power constraint.

$$C = \frac{1}{2}\log\left(1 + \frac{P}{\sigma^2}\right)$$

Setup:

- Consider an (n, M_n)-random code.
- Codebook: $\{x^n(w)\}_{w=1}^{M_n} \subset \mathbb{R}^n$
- Generate each codeword $x^n(w) = (x_1(w), \ldots, x_n(w))$ with components i.i.d. $\mathcal{N}(0, P')$, where $P' < P$ to account for power constraint smoothing.
- Encoder maps message w to $x^n(w)$.

- Channel model:
$$y^n = x^n(w) + z^n, \quad z^n \sim \mathcal{N}(0, \sigma^2 I_n)$$
- Decoder uses joint typicality:
$$A(n,\delta) = \left\{ (x^n, y^n) \in \mathbb{R}^n \times \mathbb{R}^n : \left| \frac{1}{n} \log \frac{1}{f_{X^nY^n}(x^n, y^n)} - h(X,Y) \right| < \delta \right\}$$

Covariance and Entropy Computation:

Let the joint distribution of (X, Y) be induced via the transformation:

$$Y = X + Z, \quad X \sim \mathcal{N}(0, P'), \quad Z \sim \mathcal{N}(0, \sigma^2), \text{ independent}$$

- Covariance matrix:
$$\mathrm{Cov}(X,Y) = \begin{bmatrix} \mathbb{E}[X^2] & \mathbb{E}[XY] \\ \mathbb{E}[XY] & \mathbb{E}[Y^2] \end{bmatrix} = \begin{bmatrix} P' & P' \\ P' & P' + \sigma^2 \end{bmatrix}$$
- Differential entropies:
$$\begin{aligned} h(X) &= \frac{1}{2}\log(2\pi e P') \\ h(Y) &= \frac{1}{2}\log\bigl(2\pi e(P' + \sigma^2)\bigr) \\ h(X,Y) &= \frac{1}{2}\log\left((2\pi e)^2 \cdot \det(\mathrm{Cov}(X,Y))\right) \\ &= \frac{1}{2}\log\left((2\pi e)^2 \cdot (P'(P' + \sigma^2) - P'^2)\right) \\ &= \frac{1}{2}\log\left((2\pi e)^2 P'\sigma^2\right) \end{aligned}$$
- Mutual information:
$$\begin{aligned} I(X;Y) &= h(X) + h(Y) - h(X,Y) \\ &= \frac{1}{2}\log(2\pi e P') + \frac{1}{2}\log\bigl(2\pi e(P' + \sigma^2)\bigr) - \frac{1}{2}\log\bigl((2\pi e)^2 P'\sigma^2\bigr) \\ &= \frac{1}{2}\log\left(\frac{P' + \sigma^2}{\sigma^2}\right) = \frac{1}{2}\log\left(1 + \frac{P'}{\sigma^2}\right) \end{aligned}$$

Decoding Rule:

Given received y^n, decoder declares $\hat{w}$ to be the unique index such that:

$$(x^n(\hat{w}), y^n) \in A(n, \delta)$$

If none or more than one such index exists, declare an error.

Error Analysis: (Assume $w = 1$ was sent)

- Define three error events:

  (1) $\|x^n(1)\|^2 > nP$ (violates power constraint)

  (2) $(x^n(1), y^n) \notin A(n, \delta)$

  (3) $\exists w' \neq 1 : (x^n(w'), y^n) \in A(n, \delta)$

- Use union bound:

$$\begin{aligned}\mathbb{P}_{\text{error}} \leq \mathbb{P}\left[\frac{1}{n}\sum_{i=1}^{n} x_i^2(1) > P\right] &+ \mathbb{P}\left[(x^n(1), y^n) \notin A(n, \delta)\right] \\ &+ \sum_{w' \neq 1} \mathbb{P}\left[(x^n(w'), y^n) \in A(n, \delta)\right]\end{aligned}$$

- By law of large numbers and typicality lemmas:

$$\mathbb{P}_{\text{error}} \leq \frac{\varepsilon}{6} + \frac{\varepsilon}{6} + M_n \cdot 2^{-n[I(X;Y)-3\delta]} \leq \frac{\varepsilon}{2} + M_n \cdot 2^{-n[I(X;Y)-3\delta]}$$

- Choose:

$$M_n = 2^{n[I(X;Y)-4\delta]} = 2^{n\left[\frac{1}{2}\log\left(1+\frac{P'}{\sigma^2}\right)-4\delta\right]}$$

  Then $\mathbb{P}_{\text{error}} \leq \varepsilon$ for large enough n.

Power Constraint Correction:

If some codewords exceed the power constraint P, prune them. The average probability of error per codeword remains small. Specifically:

- Prune all codewords with error probability $> \varepsilon$. This leaves at least $M'_n \geq \frac{M_n}{2}$ valid codewords.

- Then:

$$\frac{1}{n}\log M'_n \geq \frac{1}{n}\log M_n - \frac{\log 2}{n} \geq \frac{1}{2}\log\left(1 + \frac{P}{\sigma^2}\right) - 6\delta$$

  for large n and $P' \to P$.

Conclusion:

By choosing $P' \to P$ and $\delta \to 0$, the rate approaches:

$$R = \frac{1}{2}\log\left(1 + \frac{P}{\sigma^2}\right)$$

This concludes the achievability proof for the Gaussian channel with average power constraint P.

(20250401#277)

Why is the communication channel often modeled as $Y_i = X_i + Z_i$, and how does this model arise from physical principles in bandlimited signal transmission?

- The model $Y_i = X_i + Z_i$ arises naturally from physical models in communication systems, particularly when dealing with analog waveform transmission and its digital abstraction.
- A common transmission method uses bandpass modulation with carrier frequency f_c. The transmitted signals often take the form of sinusoidal carriers modulated by message signals. Typical basis functions include:

  $$\cos(2\pi f_c t), \quad \sin(2\pi f_c t)$$

  These serve as orthogonal carriers to represent information via amplitude modulation (AM), phase modulation (PM), or quadrature amplitude modulation (QAM).
- In a simple model, the **message** can be encoded as the amplitude of the signal. For instance, to transmit a bit, we might switch the amplitude of a cosine wave:

  $$\text{Bit 1: } +\cos(2\pi f_c t), \quad \text{Bit 0: } -\cos(2\pi f_c t)$$

  However, flipping the phase abruptly (e.g., from $+1$ to -1) introduces **discontinuities** in the time domain.
- These discontinuities imply that the signal is no longer a pure tone $\cos(2\pi f_c t)$ but contains additional spectral components. This creates **local perturbations** in the signal, i.e., deviations from the pure carrier due to encoding information.
- When no information is being sent, the signal would ideally be a single pure tone at frequency f_c. However, when information is encoded via modulation, this causes a **spectral spread** around f_c.
- The transmitted signal is therefore said to be **bandlimited** around the carrier frequency f_c. The message signal modulates the carrier, and the overall spectrum lies in a band centered at f_c with bandwidth determined by the message.
- Due to the real-valued nature of physical signals, the spectrum exhibits **conjugate symmetry**. That is, a band around f_c implies an identical band around $-f_c$ as well.
- To simplify analysis, especially in digital communication theory, we often apply a **frequency shift** that brings the carrier to baseband ($f_c = 0$). This yields a model where we observe the **baseband equivalent signal**.

Baseband Model and Discrete-Time Approximation

- Once the signal is represented in baseband, and after appropriate sampling (under the Nyquist criterion), the analog system is modeled as a sequence of samples $\{X_i\}$.

- The channel also adds noise, modeled as a sequence $\{Z_i\}$, typically assumed to be independent and identically distributed (i.i.d.), with a Gaussian distribution in many practical cases.
- The received signal $\{Y_i\}$ is then modeled as:

  $$Y_i = X_i + Z_i$$

  This is the standard model for an **additive noise channel**, particularly the Additive White Gaussian Noise (AWGN) channel when $Z_i \sim \mathcal{N}(0, \sigma^2)$.

(20250401#278)

Give representation of bandlimited waveforms:

Any bandlimited waveform $x(t)$ can be expressed using an orthonormal basis in terms of cosine and sine modulated sinc functions. This representation is particularly relevant in the context of passband communication systems where the signal is centered around a carrier frequency f_c. The signal takes the form:

$$x(t) = \sum_{m \in \mathbb{Z}} \left(x_{m,1}\, \phi_{m,1}^{w,f_c}(t) - x_{m,2}\, \psi_{m,2}^{w,f_c}(t) \right)$$

Here:

- $x_{m,1}$ and $x_{m,2}$ are real-valued coefficients representing the amplitude components of the waveform in the orthonormal basis.
- The functions $\phi_{m,1}^{w,f_c}(t)$ and $\psi_{m,2}^{w,f_c}(t)$ are defined as:

  $$\phi_{m,1}^{w,f_c}(t) = \sqrt{2} \cos(2\pi f_c t) \cdot W \operatorname{sinc}(W(t - m/W))$$

  $$\psi_{m,2}^{w,f_c}(t) = \sqrt{2} \sin(2\pi f_c t) \cdot W \operatorname{sinc}(W(t - m/W))$$

- These functions form an orthonormal basis for the space of real-valued, bandlimited signals centered at frequency f_c with bandwidth $2W$.
- The sinc function is defined as:

  $$\operatorname{sinc}(t) = \frac{\sin(\pi t)}{\pi t}$$

  It has zeros at all non-zero integer values, making it ideal for reconstructing signals from discrete samples.

Connection to Sampling

The above representation aligns with Shannon's Sampling Theorem, which states that any bandlimited signal (with bandwidth W) can be perfectly reconstructed from its samples taken at a rate of at least $2W$ samples per second. The sinc functions serve as interpolation kernels that ensure exact reconstruction from the sampled values.

Interpretation in Communications

- The representation allows us to decompose the passband waveform into in-phase (cosine) and quadrature (sine) components.
- Each coefficient $x_{m,1}$, $x_{m,2}$ modulates an orthonormal waveform localized in time and frequency.
- This decomposition helps analyze the behavior of the signal near the carrier frequency f_c, especially when considering real signals whose spectra exhibit conjugate symmetry.

Thus, this orthonormal expansion is crucial in understanding modulation, signal shaping, and efficient digital communication system design.

(20250401#279)

How to write a real bandlimited passband signal using complex-valued baseband signals?

A real bandlimited passband signal $x(t)$ centered around a carrier frequency f_c can be written using complex-valued baseband signals as:

$$x(t) = \mathrm{Re}\left\{ \sum_{m \in \mathbb{Z}} (x_{m,1} + jx_{m,2}) \left(\phi_{m,1}^{w,f_c}(t) + j\phi_{m,2}^{w,f_c}(t) \right) \right\}$$

Using Euler's formula and baseband modulation, this becomes:

$$x(t) = \mathrm{Re}\left\{ \sqrt{2} \sum_{m \in \mathbb{Z}} (x_{m,1} + jx_{m,2}) \,\mathrm{sinc}(w(t - m/w)) \cdot e^{j2\pi f_c t} \right\}$$

Interpretation:

- The signal is constructed by modulating a baseband signal $\sum_m \tilde{x}_m \,\mathrm{sinc}(w(t - m/w))$, where $\tilde{x}_m = x_{m,1} + jx_{m,2}$, up to a carrier frequency f_c.

- This corresponds to taking a baseband pulse and upconverting it via multiplication with the complex exponential $e^{j2\pi f_c t}$.
- The baseband representation lies entirely in the low-frequency range, and when modulated, the spectrum is centered around f_c.

Fourier Transform Viewpoint

We use the following Fourier transform pair:

$$\text{rect}\left(\frac{f}{W}\right) \overset{\mathcal{F}}{\leftrightarrow} W \cdot \text{sinc}(Wt)$$

That is, a time-domain sinc function corresponds to a frequency-domain rectangular function of bandwidth W. Hence, the sinc basis ensures the time-domain function is bandlimited.

Isometric Representation in a Hilbert Space

The continuous-time signal $x(t)$, though infinite in duration, can be equivalently represented as a countable set of real numbers — the coefficients of its projection onto an orthonormal basis. This is a mapping from continuous-time functions to infinite-dimensional real vectors:

$$x(t) \equiv \begin{pmatrix} \vdots \\ x_{-1,1} \\ x_{-1,2} \\ x_{0,1} \\ x_{0,2} \\ x_{1,1} \\ x_{1,2} \\ \vdots \end{pmatrix} \in \ell^2(\mathbb{R})$$

Similarly, any other signal such as $y(t)$ can also be represented as a vector in this infinite-dimensional Hilbert space:

$$y(t) \equiv \begin{pmatrix} \vdots \\ y_{-1,1} \\ y_{-1,2} \\ y_{0,1} \\ y_{0,2} \\ y_{1,1} \\ y_{1,2} \\ \vdots \end{pmatrix}$$

This vector space representation preserves inner products (isometry), and allows signal processing tasks (e.g., detection, filtering) to be understood as geometric operations in vector spaces.

(20250401#280)

How to represent a signal in hilbert space and how is noise analysis performed?

Let $x(t), y(t) \in L^2(\mathbb{R})$, the space of square-integrable real functions. The inner product (dot product) in this Hilbert space is defined as:

$$\langle x(\cdot), y(\cdot) \rangle = \int_{-\infty}^{\infty} x(t)y(t)\, dt$$

Using an orthonormal basis $\{\psi_{m,i}^{w,f_c}(t)\}_{m\in\mathbb{Z}, i=1,2}$, any bandlimited signal $x(t)$ can be expanded as:

$$x(t) = \sum_{m\in\mathbb{Z}} \sum_{i=1}^{2} x_{m,i}\, \psi_{m,i}^{w,f_c}(t)$$

where the coefficients are given by:

$$x_{m,i} = \langle x(\cdot), \psi_{m,i}^{w,f_c}(\cdot) \rangle$$

From Hilbert Space to Infinite-Dimensional Euclidean Space

The set of coefficients $\{x_{m,i}\}$ can be seen as coordinates in an infinite-dimensional Euclidean space ℓ^2. The inner product becomes:

$$\langle x(\cdot), y(\cdot) \rangle = \sum_{m\in\mathbb{Z}} \sum_{i=1}^{2} x_{m,i} y_{m,i} = \langle \mathbf{x}, \mathbf{y} \rangle_{\ell^2}$$

Properties of the Mapping:

- This transformation is an *isometry*: it preserves inner products.
- Norms and angles are preserved under this mapping.
- Consequently, the geometry of signal space is preserved.

Adding Noise

Suppose the received signal is corrupted by additive noise:

$$y(t) = x(t) + z(t)$$

Projecting this onto the orthonormal basis:

$$y_{m,i} = \langle y(\cdot), \psi_{m,i}^{w,f_c}(\cdot)\rangle = x_{m,i} + \underbrace{\langle z(\cdot), \psi_{m,i}^{w,f_c}(\cdot)\rangle}_{z_{m,i}}$$

So, the noise also has a representation:

$$z(t) = \sum_{m\in\mathbb{Z}} \sum_{i=1}^{2} z_{m,i}\, \psi_{m,i}^{w,f_c}(t)$$

Interpretation of Noise Components

Each $z_{m,i}$ represents the projection of noise in the direction $\psi_{m,i}^{w,f_c}(t)$, i.e., the component of the noise along that basis function.

If the noise $z(t)$ is white Gaussian noise with two-sided power spectral density $N_0/2$, then:

- $z_{m,i} \sim \mathcal{N}(0, N_0/2)$, since each projection has variance $N_0/2$.
- The coefficients $\{z_{m,i}\}$ are i.i.d. Gaussian random variables.
- The received coefficients are:

$$y_{m,i} = x_{m,i} + z_{m,i}$$

Hence, noise can be fully analyzed in this coefficient space. As the signal and noise are both represented in the same basis, the detection and decoding problem becomes an additive Gaussian noise problem in infinite-dimensional Euclidean space:

$$\mathbf{y} = \mathbf{x} + \mathbf{z}$$

Sampling Interpretation

Since the basis functions are modulated sinc functions centered at $t = \frac{m}{w}$, the coefficients $x_{m,i}$ and $y_{m,i}$ correspond to samples of the baseband signal along different orthogonal components (rails). Each rail represents either the real or imaginary (cosine or sine) component.

(20250401#281)

What is the energy of the signal at $m = 0$?

Consider a baseband signal modulated onto a passband waveform. For simplicity, analyze the energy of the component at time index $m = 0$:

$$x(t) = \sqrt{2}\,(x_{m,1}\cos(2\pi f_c t) + x_{m,2}\sin(2\pi f_c t))\,W \cdot \text{sinc}(Wt)$$

The total energy of this signal is:

$$\int_{-\infty}^{\infty} |x(t)|^2 \; dt = \int_{-\infty}^{\infty} 2\left(x_{m,1}^2 + x_{m,2}^2\right) W^2 \cdot \text{sinc}^2(Wt)\, dt$$

Using the fact that:

$$\int_{-\infty}^{\infty} W^2 \cdot \text{sinc}^2(Wt)\, dt = W$$

we get:

$$\text{Energy} = 2\left(x_{m,1}^2 + x_{m,2}^2\right) \cdot W$$

So, the energy per symbol interval $1/W$ becomes:

$$E = \left(x_{m,1}^2 + x_{m,2}^2\right)$$

This is consistent with interpreting $x_{m,1}$ and $x_{m,2}$ as real-valued components (in-phase and quadrature) modulating orthogonal carriers.

Dimensionality of the Signal

Each interval of length $1/W$ allows us to transmit a single complex value, or equivalently, two real-valued symbols. Over a duration T, we thus have:

- WT complex degrees of freedom.
- $2WT$ real degrees of freedom.

Average Signal Power

The total signal energy over time T is:

$$\sum_{m=1}^{WT} \left(x_{m,1}^2 + x_{m,2}^2\right)$$

Then, the **average signal power** is:

$$P = \frac{1}{T} \sum_{m=1}^{WT} \left(x_{m,1}^2 + x_{m,2}^2 \right) \quad \text{(units: Watts)}$$

The average power **per real dimension** is:

$$\frac{1}{2WT} \sum_{m=1}^{WT} \left(x_{m,1}^2 + x_{m,2}^2 \right) = \frac{P}{2W}$$

Noise Model and Signal-to-Noise Ratio (SNR)

Assume white Gaussian noise $z(t)$ with two-sided power spectral density $N_0/2$. Then:

- Noise power per real dimension is $N_0/2$
- Signal power per real dimension is $P/(2W)$

Therefore, the signal-to-noise ratio (SNR) per real dimension is:

$$\text{SNR} = \frac{P/(2W)}{N_0/2} = \frac{P}{N_0 W}$$

This SNR formulation will play a central role in analyzing the capacity of the bandlimited Gaussian channel.

(20250401#282)

Obtain the number of dimensions, messages and capacity in a bandlimited AWGN channel:

We consider a communication system with:

- Bandwidth constraint: W Hz.
- Power constraint: $\overline{P}$ Watts (i.e., $\overline{P}$ joules/second).
- Additive White Gaussian Noise (AWGN): $z(t)$, with power spectral density $N_0/2$ Watts/Hz.
- Received signal: $y(t) = x(t) + z(t)$.

Dimensionality of the Signal

Under a bandwidth constraint W, over a time duration T, the number of complex degrees of freedom (DoF) is:

$$\text{Number of complex DoF} = WT$$

Equivalently, this gives:

$$\text{Number of real dimensions} = 2WT$$

These dimensions can be thought of as coordinates in a $2WT$-dimensional Euclidean space, where the geometry of the signal is preserved (thanks to isometry from Hilbert space).

Capacity and Number of Messages

From Shannon's capacity formula for the AWGN channel:

$$C = W \log_2 \left(1 + \frac{\overline{P}}{N_0 W}\right) \quad \text{bits/sec}$$

The total number of messages M_n that can be reliably transmitted over duration T is:

$$\log_2 M_n = C \cdot T = WT \log_2 \left(1 + \frac{\overline{P}}{N_0 W}\right)$$

This implies:

$$\frac{\log_2 M_n}{T} = W \log_2 \left(1 + \frac{\overline{P}}{N_0 W}\right) \quad \text{bits/sec}$$

Interpretation

- The capacity depends linearly on the bandwidth W, but logarithmically on the signal-to-noise ratio $\overline{P}/(N_0 W)$.
- Increasing bandwidth increases the number of available dimensions.
- Power per dimension becomes smaller as bandwidth increases (because total power is fixed), but the increase in DoF may still yield higher capacity.
- The formula represents the maximum achievable rate (in bits per second) under power and bandwidth constraints with vanishing probability of error.

Summary

$$\boxed{\begin{aligned} \text{Dimensions} &= 2WT \text{ (real)}, \quad WT \text{ (complex)} \\ \log_2 M_n &= WT \log_2\left(1 + \frac{\overline{P}}{N_0 W}\right) \\ \text{Rate (bits/sec)} &= W \log_2\left(1 + \frac{\overline{P}}{N_0 W}\right) \end{aligned}}$$

This is the celebrated formula for the capacity of a bandlimited AWGN channel.

(20250402#283)

State the channel capacity formula for an additive white Gaussian channel:

The channel capacity C for an additive white Gaussian noise (AWGN) channel is given by Shannon's formula:

$$C = W \log_2 \left(1 + \frac{\overline{P}}{N_0 W}\right) \quad \text{bits per second}$$

where:

- W is the **bandwidth** (in Hz)
- $\overline{P}$ is the **average transmitted power** (in Watts)
- N_0 is the **noise spectral density** (in watts/Hz)

(20250402#284)

What can be said about multiplicative gain, logarithmic effect of SNR, and the limiting factors of capacity in low SNR vs high SNR cases?

---

1. **Multiplicative Effect of Bandwidth (W)**:
   - Bandwidth appears as a *linear multiplicative factor*
   - Doubling W would approximately double C (assuming SNR remains constant)
   - This represents a scaling benefit of increasing bandwidth
2. **Logarithmic Effect of Power-to-Noise Ratio ($\frac{\overline{P}}{N_0 W}$)**:
   - The $\log_2(1 + \text{SNR})$ term shows *diminishing returns*:

   $$\text{When } \frac{\overline{P}}{N_0 W} \gg 1, \quad \log_2\left(1 + \frac{\overline{P}}{N_0 W}\right) \approx \log_2\left(\frac{\overline{P}}{N_0 W}\right)$$

   - Doubling power doesn't double capacity:

   $$\text{If } \frac{\overline{P}}{N_0 W} \to 2\frac{\overline{P}}{N_0 W}, \quad C \to W \log_2(2) + C_{\text{original}} = W + C_{\text{original}}$$

   - This shows only *additive* rather than multiplicative improvement
3. **Power-Limited vs Bandwidth-Limited Regimes**:

- **Low SNR** ($\frac{\overline{P}}{N_0 W} \ll 1$): Power is limiting factor

$$\log_2(1+x) \approx x \log_2 e \quad \Rightarrow \quad C \approx \frac{\overline{P}}{N_0} \log_2 e$$

- **High SNR** ($\frac{\overline{P}}{N_0 W} \gg 1$): Bandwidth is limiting factor

$$C \approx W \log_2 \left(\frac{\overline{P}}{N_0 W} \right)$$

(20250402#285)

How should the SNR vary to increase the capacity by 1 bit?

$$\begin{aligned}
\text{When SNR} = 1 &\quad \Rightarrow \log_2(2) = 1 \text{ bit} \\
\text{SNR} = 3 &\quad \Rightarrow \log_2(4) = 2 \text{ bits} \\
\text{SNR} = 7 &\quad \Rightarrow \log_2(8) = 3 \text{ bits} \\
&\vdots \\
\text{SNR} = 2^n - 1 &\Rightarrow n \text{ bits}
\end{aligned}$$

Each additional bit requires *exponentially increasing power*.

(20250402#286)

Give the formula for spectral efficiency. Why is it called the spectral efficiency?

Spectral efficiency:

$$\frac{C}{W} \ \text{bits/s/Hz} = \log \left(1 + \frac{\overline{P}}{N_0 W} \right)$$

The term **spectral efficiency** originates from how effectively a communication system utilizes its allocated frequency spectrum. Mathematically, it measures how many bits can be transmitted per unit bandwidth:

$$\eta \triangleq \frac{C}{W} \quad \text{(bits/s/Hz)}$$

where:

- C is the channel capacity (bits/s)

- W is the bandwidth (Hz)

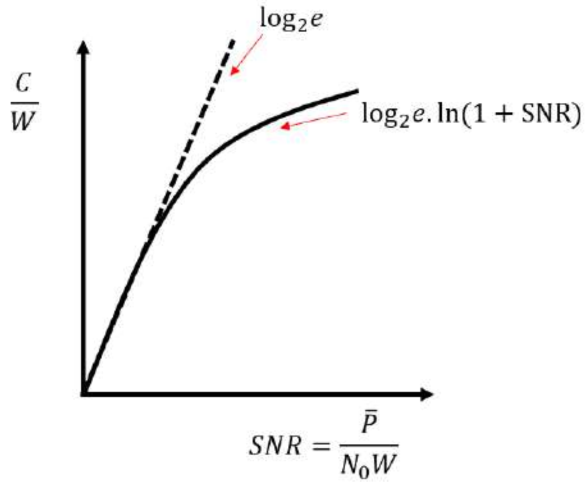
For an AWGN channel with SNR γ:

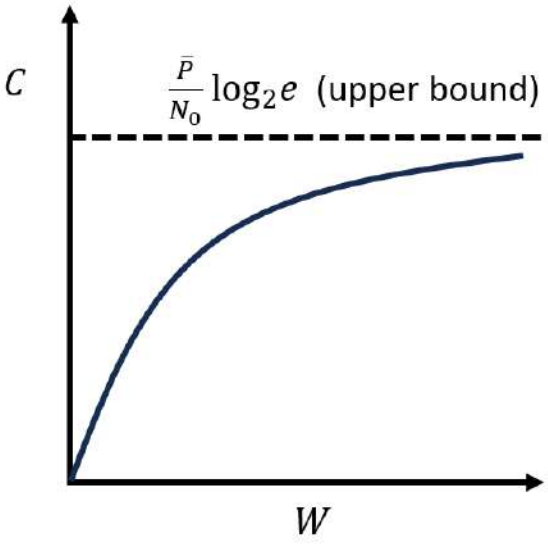
$$\eta = \log_2(1+\gamma)$$

This shows: **Inefficiency** appears as the log term's sublinear growth

(20250402#287)

Plot spectral efficiency as a function of SNR. Also plot capacity C as a function of bandwidth.





In the plot for C/W vs SNR, initially, when SNR is small, we use the formula $\log(1+x) \approx x$ to come up with $\log_2 e$ kind of variation for C/W with SNR. Then because of diminishing returns, as SNR is inside the log function, the value of the function will deviate (slower rate of increase as compared to the $\log_2 e$) away from $\log_2 e$.

The upperbound in the figure for C vs W is obtained as follows: As W becomes very large, $\overline{P}/N_0W$ becomes very small. Then we can use $\ln(1+x) \approx x$ assumption, which will give us

$$C = W\log\left(1+\frac{\overline{P}}{N_0W}\right) \approx W\frac{\overline{P}}{N_0W}\log_2 e = \frac{\overline{P}}{N_0}\log_2 e$$

The behavior of C vs W follows from this:

$$\begin{aligned} C &= W\log\left(1+\frac{\overline{P}}{N_0W}\right) \\ &= \log\left(1+\frac{\overline{P}}{N_0W}\right)/\frac{1}{W} \end{aligned}$$

It is like $\log(1+x)/x$ which is decreasing in x. But x axis is W while x here is $1/W$. So with W, the function $C(W)$ increases.

(20250402#288)

Give the expression for energy per bit E_b. What does it E_b/N_0 mean?

Energy per bit E_b is given by

$$E_b = \frac{\overline{P}\cancel{T}\text{ J}}{\cancel{T}W\log\left(1+\frac{\overline{P}}{N_0W}\right)}$$

People look at E_b in relation to the noise level. So typically we see E_b/N_0. For the maximum capacity, we'll have $(E_b/N_0)_{min}$.

$$\frac{E_b}{N_0} = \left(\frac{\overline{P}/N_0W}{\log\left(1+\overline{P}/N_0W\right)}\right)_{\text{when minimized}}$$

Minimum E_b/N_0 can be thought of as the minimum energy spent per bit. To achieve this, one must spread the signal across a huge bandwidth such that

$$\frac{\overline{P}/2W}{N_0/2} \to 0$$

(20250402#289)

Obtain the value for minimum energy per bit in decibels.

$(E_b/N_0)_{min}$ is typically expressed in decibels. It is least when $\overline{P}/N_0W = 0$. As $\overline{P}/N_0W \to 0$, we have

$$\frac{E_b}{N_0} \to \frac{1}{\log_2 e} = \ln 2$$

→ spread your signal across the bandwidth as much as possible to get minimum energy per bit. Thus in decibels, $10\log_{10}(\ln 2)$ dB $= 10\log_{10}(0.693)$ dB $= -1.59$ dB

(20250402#290)

What are parallel gaussian channels?

Parallel gaussian channels: A set of K **independent** Gaussian channels operating in parallel, where each sub-channel k has:

$$Y_k = X_k + Z_k, \quad k = 1, 2, \dots, K$$

with:

- X_k: Input signal (power constraint $\mathbb{E}[X_k^2] \leq P_k$)
- $Z_k \sim \mathcal{N}(0, N_k)$: Additive Gaussian noise
- Y_k: Output signal

(20250402#291)

What is a vector gaussian channel?

A d-dimensional vector gaussian channel (in discrete-time) is defined by:

$$y_i = x_i + z_i$$

where:

- $x_i \in \mathbb{R}^d$ is the input vector with the power constraint

$$\frac{1}{n}\|x_i^2\| \leq P$$

- $\boldsymbol{z_i} \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$: Additive gaussian noise vector, $(z_i)_{i\geq 1}$ is *iid*.
- $\boldsymbol{y_i} \in \mathbb{R}^d$: Output vector

The covariance matrix $\mathbf{\Sigma}$ is diagonal with independent noise components:

$$\mathbf{\Sigma} = \begin{pmatrix} \sigma_1^2 & 0 & \cdots & 0 \\ 0 & \sigma_2^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_d^2 \end{pmatrix}$$

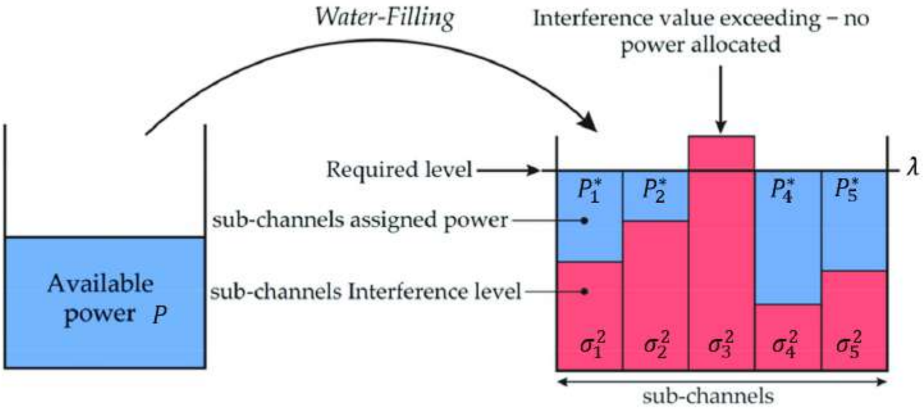
This implies:

- Noise components are **mutually independent** but not identically distributed.
- i-th dimension has noise variance σ_i^2
- Equivalent to d parallel scalar Gaussian channels

(20250402#292)

How is capacity achieved in the vector gaussian channel?

The power is distributed among the d-dimensions via waterfilling method. The capacity achieved would be



$$C = \sum_{j=1}^{d} \frac{1}{2} \log_2 \left(1 + \frac{P_j^*}{\sigma_j^2}\right)$$

where P_j is the optimal power allocation (water-filling solution):

$$P_j^* = \begin{cases} \lambda - \sigma_j^2 & \text{if } \lambda > \sigma_j^2 \\ 0 & \text{otherwise} \end{cases}$$

where we've used truncation to 0, if σ_k^2 exceeds λ for any $k \in \{1, 2, 3, \ldots, d\}$. We denote it as

$$P_j^* = \left[\lambda - \sigma_j^2\right]_+, \text{ where } j = 1, 2, 3, \ldots, d.$$

with λ chosen to satisfy $\sum_{j=1}^d P_j^* = \sum_{j=1}^d [\lambda - \sigma_j^2]_+ = P$.

(20250402#293)

Obtain the expression for capacity of a d-dimensional vector gaussian channel with isotropic noise. What parallel channel is this equivalent to?

- **Isotropic Noise** ($\sigma_i^2 = \sigma^2 \ \forall i$):

$$C = \frac{d}{2}\log_2\left(1 + \frac{P}{d\sigma^2}\right)$$

- **Identical Parallel Channels** ($\sigma_i^2 = \sigma^2$, equal power allocation):

$$C = \frac{d}{2}\log_2\left(1 + \frac{P}{d\sigma^2}\right)$$

(20250402#294)

Give an intuition as to why water filling is the right approach:

The water-filling approach is optimal for power allocation in a vector Gaussian channel. Consider channel j with optimal power $P_j^* > 0$ and another channel $j' \neq j$ with power $P_{j'}^*$.

Perform a perturbation analysis: move a small δ power from j to j'. Using first-order Taylor approximation, the derivative of the capacity

$$C = \frac{1}{2}\sum_{j=1}^{d}\log\left(1 + P_j^*/\sigma_j^2\right)$$

yields the condition:

$$\frac{-\delta \log_2 e}{\sigma_j^2 + P_j^*} + \frac{\delta \log_2 e}{\sigma_j^2 + P_{j'}^*} \leq 0$$

Any deviation from optimal allocation decreases capacity.

If $\frac{1}{\sigma_j^2+P_j^*} < \frac{1}{\sigma_j^2+P_{j'}^*}$, moving power to j' would improve capacity. Thus, at optimality:

$$\frac{1}{\sigma_j^2 + P_j^*} \geq \frac{1}{\sigma_j^2 + P_{j'}^*}$$

This implies for all j with $P_j^* > 0$:

$$\sigma_j^2 + P_j^* = \lambda$$

where λ is the water level. For inactive channels ($P_j^* = 0$):

$$\frac{1}{\sigma_j^2} < \frac{1}{\lambda} \quad \Leftrightarrow \quad \lambda < \sigma_j^2$$

In depth derivation of optimality condition: Consider the capacity function:

$$C(\mathbf{P}) = \frac{1}{2} \sum_{i=1}^{d} \log_2 \left(1 + \frac{P_i}{\sigma_i^2}\right)$$

Let $\mathbf{P}^*$ be the optimal power allocation. We perturb by moving δ power from channel j to j', creating new allocation:

$$P_j = P_j^* - \delta, \quad P_{j'} = P_{j'}^* + \delta$$

The capacity change is:

$$\Delta C = C(\mathbf{P}) - C(\mathbf{P}^*) = \frac{1}{2} \left[\log_2 \left(1 + \frac{P_j^* - \delta}{\sigma_j^2}\right) + \log_2 \left(1 + \frac{P_{j'}^* + \delta}{\sigma_j'^2}\right)\right] - \text{original terms}$$

Using Taylor expansion around $\delta = 0$:

$$\log_2 \left(1 + \frac{P_j^* - \delta}{\sigma_j^2}\right) \approx \log_2 \left(1 + \frac{P_j^*}{\sigma_j^2}\right) - \frac{\delta}{\sigma_j^2 + P_j^*} \log_2 e$$

$$\log_2 \left(1 + \frac{P_{j'}^* + \delta}{\sigma_{j'}^2}\right) \approx \log_2 \left(1 + \frac{P_{j'}^*}{\sigma_{j'}^2}\right) + \frac{\delta}{\sigma_{j'}^2 + P_{j'}^*} \log_2 e$$

Substituting back:

$$\Delta C \approx \frac{1}{2} \left[-\frac{\delta}{\sigma_j^2 + P_j^*} \log_2 e + \frac{\delta}{\sigma_{j'}^2 + P_{j'}^*} \log_2 e\right]$$

For optimality, any perturbation must not increase capacity ($\Delta C \leq 0$):

$$-\frac{1}{\sigma_j^2 + P_j^*} + \frac{1}{\sigma_{j'}^2 + P_{j'}^*} \leq 0$$

$$\Rightarrow \frac{1}{\sigma_j^2 + P_j^*} \geq \frac{1}{\sigma_{j'}^2 + P_{j'}^*}$$

Equality holds when both channels are active ($P_j^*, P_{j'}^* > 0$), leading to the water-filling condition:

$$\sigma_j^2 + P_j^* = \sigma_{j'}^2 + P_{j'}^* = \lambda$$

(20250402#295)

Give a proof for the waterfilling approach:

Consider a vector Gaussian channel with d parallel subchannels, where:

- σ_i^2 is the noise variance in subchannel i
- P_i is the power allocated to subchannel i
- Total power constraint: $\sum_{i=1}^d P_i \leq P_{\text{total}}$

The channel capacity is:

$$C = \max_{\{P_i\}} \frac{1}{2} \sum_{i=1}^{d} \log_2 \left(1 + \frac{P_i}{\sigma_i^2}\right) \quad \text{subject to} \quad P_i \geq 0, \sum_{i=1}^{d} P_i \leq P_{\text{total}}$$

Proof of Waterfilling Optimality:

1. **Lagrangian Formulation**:

$$\mathcal{L} = \frac{1}{2} \sum_{i=1}^{d} \log_2 \left(1 + \frac{P_i}{\sigma_i^2}\right) - \lambda \left(\sum_{i=1}^{d} P_i - P_{\text{total}}\right) + \sum_{i=1}^{d} \mu_i P_i$$

2. **Karush-Kuhn-Tucker (KKT) Conditions**:

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial P_i} &= \frac{1}{2} \frac{1}{\sigma_i^2 + P_i} \log_2 e - \lambda + \mu_i = 0 \\ \mu_i P_i &= 0 \quad \text{(Complementary slackness)} \\ P_i &\geq 0, \quad \mu_i \geq 0 \end{aligned}$$

3. **Case Analysis**:

- For $P_i > 0$ (active subchannels), $\mu_i = 0$:

$$\frac{1}{\sigma_i^2 + P_i} = 2\lambda / \log_2 e \implies P_i = \left(\frac{\log_2 e}{2\lambda} - \sigma_i^2\right)^+$$

- For $P_i = 0$ (inactive subchannels):

$$\frac{1}{\sigma_i^2} \leq 2\lambda / \log_2 e \implies \sigma_i^2 \geq \frac{\log_2 e}{2\lambda}$$

4. **Waterfilling Interpretation**:

- Define water level $\nu = \frac{\log_2 e}{2\lambda}$
- Optimal power allocation:

$$P_i^* = \begin{cases} \nu - \sigma_i^2 & \text{if } \sigma_i^2 < \nu \\ 0 & \text{otherwise} \end{cases}$$

- ν is chosen such that $\sum_{i=1}^{d} P_i^* = P_{\text{total}}$

5. **Uniqueness**: The capacity function is strictly concave in $\{P_i\}$ and the power constraint defines a convex set, so the waterfilling solution is the unique global maximum.

Geometric Interpretation: The solution "pours" power into the channels until:

$$\sigma_i^2 + P_i = \text{constant} = \nu \quad \text{(water level)}$$

for all active channels, with weaker channels (higher σ_i^2) receiving less or no power.

(20250402#296)

What is the general approach for finding the capacity of a channel with colored noise?

For colored noise $\mathbf{z}$ with covariance $\mathcal{K}$:

$$\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathcal{K})$$

whiten the noise through change of basis and using Karhunen-Loeve transform. Then proceed as we did before for AWGN.

Whitening Procedure:

1. **Eigen-decomposition** of covariance matrix:

$$\mathcal{K} = U \Lambda U^T$$

where:

- U is orthogonal ($U^T = U^{-1}$)
- Λ is diagonal (possibly singular)
- For singular $\mathcal{K}$, Λ has some zero eigenvalues

2. **Transform the observation y**:

$$\hat{\mathbf{y}} = U^T\mathbf{y}$$

$$\hat{\boldsymbol{y}} = U^T\mathbf{y} = U^T\mathbf{x} + U^T\mathbf{z}$$

We'll call $\hat{\boldsymbol{z}} = U^T\mathbf{z}$

$$\|\hat{x}\|^2 = \hat{x}^T\hat{x} = (U^Tx)^T(U^Tx)$$

Using the property of matrix transposes:

$$= x^TUU^Tx$$

Since U is orthogonal ($U^TU = UU^T = I_d$):

$$= x^TIx = x^Tx = \|x\|^2$$

Geometric Interpretation:

Orthogonal transformations preserve:

- Vector lengths ($\|U^Tx\| = \|x\|$)
- Angles between vectors ($\langle U^Tx, U^Ty\rangle = \langle x, y\rangle$)
- The identity follows from $\|x\|^2 = \langle x, x\rangle$

3. **Properties**:

- **Information Preservation**:

$$UU^T = U^TU = I_d \quad \Rightarrow \quad \text{Transformation is invertible}$$

No information is lost in forward/backward transform.

- **Whitening Effect**:

$$\mathbb{E}[\hat{\mathbf{z}}\hat{\mathbf{z}}^T] = U^T\mathcal{K}U = \Lambda \quad \text{(Diagonal covariance)}$$

- **Singular Case Handling**: When $\mathcal{K}$ is positive semi-definite (some $\lambda_i = 0$):

$$\hat{y}_i = 0 \text{ a.s. for } \lambda_i = 0$$

These dimensions can be discarded without information loss.

Key Observations:

1. The transform projects the signal into the eigenbasis where noise components are uncorrelated.

2. For non-singular $\mathcal{K}$:
$$\Lambda^{-1/2}U^T \text{ yields spherical noise}$$

3. For singular $\mathcal{K}$, the operation effectively:
$$\text{1) Rotates, 2) Scales, 3) Projects out noise-free dimensions}$$

(20250402#297)

Why does the noise concentrate along the semi-major axis of the ellipsoid?

Consider a zero-mean Gaussian noise vector $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathcal{K})$ with covariance matrix $\mathcal{K}$.

Geometry of the Noise Distribution:

1. The equiprobability contours form ellipsoids:
$$\mathbf{z}^T\mathcal{K}^{-1}\mathbf{z} = \text{constant}$$

2. Eigen-decomposition of $\mathcal{K}$ reveals principal axes:
$$\mathcal{K} = U\Lambda U^T \quad \text{where} \quad \Lambda = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_d)$$

Why Noise Concentrates Along the Semi-Major Axis:

1. The semi-axes lengths are proportional to $\sqrt{\lambda_i}$:
   - Largest $\lambda_i \Rightarrow$ longest semi-major axis
   - Smallest $\lambda_i \Rightarrow$ shortest semi-minor axis

2. Probability density falls off exponentially with:
$$f(\mathbf{z}) \propto \exp\left(-\frac{1}{2}\sum_{i=1}^{d}\frac{(u_i^T\mathbf{z})^2}{\lambda_i}\right)$$

3. Along the semi-major axis (direction of largest $\lambda_{\max}$):

- The constraint is weakest (largest denominator)
- Noise can deviate farther while maintaining significant probability

4. Quantitatively, for standard normal $v_i = u_i^T \mathbf{z}/\sqrt{\lambda_i}$:

$$\mathbb{E}[\|v_i\|^2] = 1 \quad \Rightarrow \quad \mathbb{E}[(u_i^T \mathbf{z})^2] = \lambda_i$$

Visualization:

- 2D case: The noise ellipse stretches furthest in the direction of $\lambda_1 > \lambda_2$
- Most samples fall within the elongated region parallel to the major axis
- Minor axis directions show much faster probability decay

(20250402#298)

What is a multipath channel?

A **multipath channel** is a communication channel where signals propagate from transmitter to receiver through multiple paths due to:

- **Reflections** from buildings, mountains, etc.
- **Diffractions** around obstacles
- **Scattering** from rough surfaces

This is an example case for colorned noise vector gaussian channel scenario.

Mathematical Model

The channel impulse response is:

$$h(t) = \sum_{k=1}^{N} a_k(t) e^{j\phi_k(t)} \delta(t - \tau_k(t))$$

where:

- $a_k(t)$: Time-varying amplitude of k-th path
- $\phi_k(t)$: Time-varying phase of k-th path
- $\tau_k(t)$: Time-varying delay of k-th path

- N: Number of propagation paths

Key Effects

1. **Frequency-Selective Fading**:

$$H(f) = \mathcal{F}\{h(t)\} = \sum_{k=1}^{N} a_k e^{j\phi_k} e^{-j2\pi f \tau_k}$$

2. **Delay Spread**:

$$\tau_{\max} = \max_k \tau_k - \min_k \tau_k$$

3. **Doppler Spread** (for mobile channels):

$$f_D = \frac{v}{\lambda} \cos\theta$$

Types

- **Time-Invariant**: Fixed τ_k, a_k, ϕ_k
- **Time-Variant**: Parameters change with time
- **Flat Fading**: $\tau_{\max} \ll$ symbol period
- **Frequency-Selective**: $\tau_{\max} >$ symbol period

(20250402#299)

For colored noise, write the expressions for autocorrelation and power spectral density:

1. Autocorrelation Function:

Let $\{z_i\}_{i\geq 1}$ be a stationary colored noise process. The autocorrelation function is defined as:

$$R_z(k) = \mathbb{E}[z_i z_{i-k}]$$

For colored noise, $R_z(k) \neq 0$ for $k \neq 0$, indicating temporal correlation between samples.

2. Power Spectral Density (PSD):

The power spectral density is the Fourier transform of the autocorrelation function:

$$S_z(f) = \sum_{k=-\infty}^{\infty} R_z(k) e^{-j2\pi f k}$$

This describes how the noise power is distributed over frequency. For colored noise, the PSD is not constant.

3. Examples:

White Noise:

$$R_z(k) = \sigma^2\delta(k) \quad \Rightarrow \quad S_z(f) = \sigma^2 \text{ (flat)}$$

For z_i i.i.d., white gaussian noise gives R_k as a delta function with value σ^2 at $k = 0$ and 0 everywhere else. The spectral power density will be the same for all the frequencies in the range $[-1/2, 1/2]$.

AR(1) Noise Model: $Z_n = \alpha Z_{n-1} + W_n$, where $W_n \sim \mathcal{N}(0, \sigma_W^2)$ is white noise.

- Autocorrelation:

$$R_Z(k) = \frac{\sigma_W^2}{1-\alpha^2} \cdot \alpha^{|k|}$$

- Power Spectral Density:

$$S_Z(f) = \frac{\sigma_W^2}{|1 - \alpha e^{-j2\pi f}|^2}$$

(20250402#300)

What will be the capacity of a channel with colored noise? Give a practical scenario where this may happen:

Channel Model: Consider a discrete-time linear time-invariant (LTI) channel with memory:

$$Y_n = \sum_{k=0}^{L} h_k X_{n-k} + Z_n,$$

where $\{h_k\}$ is the impulse response of the channel (i.e., convolution with input X_n), and $\{Z_n\}$ is colored Gaussian noise, modeled as a stationary process with power spectral density (PSD) $S_Z(f)$.

Let $H(f)$ denote the frequency response of the channel:

$$H(f) = \sum_{k=0}^{L} h_k e^{-j2\pi fk}.$$

Prewhitening Approach:

We apply a whitening filter $H^{-1}(f)$ (equalizer) to make the noise white, converting the colored noise into white noise. The system then becomes:

$$\tilde{Y}_n = \tilde{H}(f)X_n + W_n,$$

where $\tilde{H}(f) = H(f)/H_Z(f)$, and W_n is white Gaussian noise.

Capacity Expression:

The capacity of the channel with colored noise is:

$$C = \int_{-1/2}^{1/2} \frac{1}{2}\log_2\left(1 + \frac{S_X(f)|H(f)|^2}{S_Z(f)}\right) df,$$

subject to the power constraint:

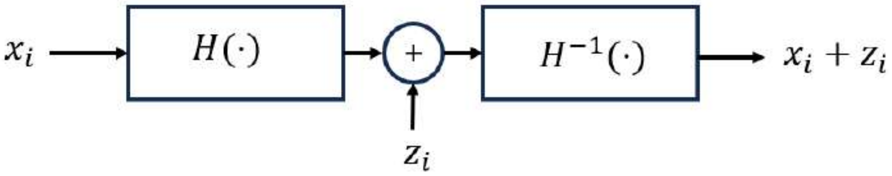
$$\int_{-1/2}^{1/2} S_X(f)df \leq P.$$

Water-filling Solution:

The optimal input PSD $S_X(f)$ is found using water-filling over the noise spectrum:

$$S_X(f) = \left(\mu - \frac{S_Z(f)}{|H(f)|^2}\right)^+,$$

where μ is chosen to satisfy the power constraint. **Practical Significance:**



This setup arises in many real-world situations, especially in:

- **Wireless Communications:** Due to multipath propagation, the received signal is a superposition of delayed and attenuated versions of the transmitted signal, modeled by a linear filter $H(f)$. Thermal and interference noise are often colored.
- **DSL/Broadband over Copper:** Crosstalk between twisted pairs introduces colored noise, and the channel impulse response includes memory.
- **Underwater Acoustic Channels:** Where reflections and Doppler spreading lead to highly colored noise environments and long memory in the channel.

Equalization: In practice, equalizers such as MMSE or ZF (zero-forcing) are used to mitigate inter-symbol interference and whiten the noise, making capacity-approaching performance possible with appropriate coding.

(20250402#301)

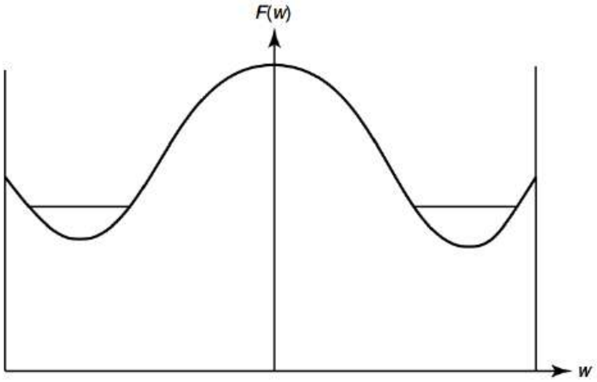
How does waterfilling in the case of colored noise continuous channel look like?

Consider a continuous-time channel with colored Gaussian noise:

$$Y(f) = H(f)X(f) + Z(f),$$

where:

- $H(f)$ is the channel frequency response,
- $X(f)$ is the input signal in the frequency domain,
- $Z(f)$ is additive Gaussian noise with power spectral density $S_Z(f)$,
- $f \in [-1/2, 1/2]$.



The capacity of this channel under average power constraint P is:

$$C = \int_{-1/2}^{1/2} \frac{1}{2} \log_2 \left(1 + \frac{S_X(f)|H(f)|^2}{S_Z(f)}\right) df,$$

subject to:

$$\int_{-1/2}^{1/2} S_X(f) df \leq P.$$

Waterfilling Power Allocation

The optimal input power spectral density is given by:

$$S_X(f) = \left(\mu - \frac{S_Z(f)}{|H(f)|^2}\right)^+,$$

where μ is chosen such that:

$$\int_{-1/2}^{1/2} \left(\mu - \frac{S_Z(f)}{|H(f)|^2}\right)^+ df = P.$$

Here, $(\cdot)^+$ denotes $\max\{0, \cdot\}$, meaning power is only allocated where the effective noise $S_Z(f)/|H(f)|^2$ is below the water level μ.

Special Case: AWGN Channel

If $H(f) = 1$ and $S_Z(f) = N_0$ (white Gaussian noise), then:

$$S_X(f) = \mu - N_0,$$

and power is uniformly allocated across the bandwidth.

(20250402#302)

Give the proof idea for continuous colored noise channel capacity:

We aim to compute the capacity of a continuous-time Gaussian channel with colored noise, where the noise process has memory (i.e., is correlated). A natural approach is to break the signal into blocks and treat each block as a high-dimensional vector. This method approximates the colored noise channel as a sequence of finite-dimensional vector Gaussian channels with memory.

Step 1: Block Structure and Decorrelation

Partition the time domain into blocks of length d, separated by finite-sized gaps. Each block is used for transmission, while the gaps act as buffers to reduce correlation between consecutive blocks.

$$(1, \ldots, d) \text{ (gap) } (d+1, \ldots, 2d) \text{ (gap) } \ldots$$

Let the gaps be large enough so that the correlation between samples from different blocks is negligible. The idea is that if the autocorrelation decays fast enough, the influence of the past/future on the current block is asymptotically small. We compensate for the power loss in the gaps by increasing the block size $d \to \infty$, so the gap fraction becomes negligible.

Step 2: Power Constraint

Each d-dimensional block $\boldsymbol{x} = (x_1, \ldots, x_d)$ must satisfy the average power constraint:

$$\frac{1}{d} \sum_{i=1}^{d} x_i^2 \leq P.$$

Or in vector notation:

$$\|\boldsymbol{x}\|^2 \leq dP.$$

Step 3: Capacity per Block

For each block of length d, the noise is modeled as a Gaussian vector with mean zero and a $d \times d$ covariance matrix $\mathcal{K}_d$, which is Toeplitz:

$$\mathcal{K}_d = (r_{i-j})_{1 \le i,j \le d}, \quad \text{with autocorrelation } r_k = \mathbb{E}[Z_i Z_{i+k}].$$

The capacity per block (in nats) is given by:

$$C_d = \max_{P_{\boldsymbol{X}} : \mathbb{E}[\|\boldsymbol{X}\|^2] \le dP} I(\boldsymbol{X}; \boldsymbol{Y}),$$

where $\boldsymbol{Y} = \boldsymbol{X} + \boldsymbol{Z}$, and $\boldsymbol{Z} \sim \mathcal{N}(0, \mathcal{K}_d)$.

This is the classic vector Gaussian channel with colored noise. The optimal input $\boldsymbol{X}$ is Gaussian with covariance matrix aligned with the eigenvectors of $\mathcal{K}_d$.

Step 4: Total Capacity

Define the capacity per channel use as:

$$C = \lim_{d \to \infty} \frac{C_d}{d}.$$

Using Szegő's theorem, we can relate the eigenvalues of $\mathcal{K}_d$ to the power spectral density $S_Z(f)$, and obtain the water-filling solution:

$$C = \int_{-1/2}^{1/2} \frac{1}{2} \log\left(1 + \frac{S_X(f)}{S_Z(f)}\right) df,$$

with the optimal input spectrum $S_X(f)$ found via water-filling under the power constraint:

$$\int_{-1/2}^{1/2} S_X(f) df \le P.$$

(20250402#303)

Lemma: State the Toeplitz distribution theorem (Szego, 1955)

Let $\{r_k\}_{k \in \mathbb{Z}}$ be a sequence of real numbers such that the corresponding Toeplitz matrix

$$\mathcal{K}_n = \begin{bmatrix} r_0 & r_1 & r_2 & \cdots & r_{n-1} \\ r_1 & r_0 & r_1 & \cdots & r_{n-2} \\ r_2 & r_1 & r_0 & \cdots & r_{n-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{n-1} & r_{n-2} & r_{n-3} & \cdots & r_0 \end{bmatrix}$$

is positive semi-definite for all n. Assume the autocorrelation sequence $\{r_k\}$ is absolutely summable:

$$\sum_{k=-\infty}^{\infty} |r_k| < \infty.$$

Define the power spectral density (PSD) as

$$S(\omega) = \sum_{k=-\infty}^{\infty} r_k e^{-j2\pi\omega k}, \quad \omega \in [-\tfrac{1}{2}, \tfrac{1}{2}].$$

Let $\lambda_1^{(n)}, \lambda_2^{(n)}, \ldots, \lambda_n^{(n)}$ be the eigenvalues of $\mathcal{K}_n$. Then, for any continuous function f,

$$\lim_{n\to\infty} \frac{1}{n} \sum_{k=1}^{n} f\left(\lambda_k^{(n)}\right) = \int_{-1/2}^{1/2} f(S(\omega))\, d\omega.$$

(20250402#304)

What does positive semi-definite matrix mean?

A matrix $A \in \mathbb{R}^{n\times n}$ is said to be **positive semi-definite (PSD)** if it satisfies the following condition:

$$\forall \mathbf{x} \in \mathbb{R}^n, \quad \mathbf{x}^\top A\mathbf{x} \geq 0$$

This means that for every real vector $\mathbf{x}$, the quadratic form $\mathbf{x}^\top A\mathbf{x}$ is non-negative.

Equivalent characterizations

- All eigenvalues of A are non-negative.
- A is symmetric (i.e., $A = A^\top$) and satisfies the quadratic form condition above.
- There exists a matrix B such that $A = B^\top B$.

Remarks

- Positive semi-definiteness of a covariance matrix ensures non-negative variances and valid correlation structures.

(20250402#305)

In the proof sketch for gaussian colored noise channel capacity, why does the Fourier terms come into the picture?

1. Colored Gaussian Noise Let the noise process $Z = (Z_1, Z_2, \ldots, Z_n)$ be a stationary Gaussian process with mean zero and autocovariance function

$$R(k) = \mathbb{E}[Z_i Z_{i+k}]$$

This induces a Toeplitz covariance matrix $\mathcal{K}_n$ on Z^n, with entries depending only on $|i - j|$.

2. Spectral Representation By the Wiener–Khinchin theorem, the autocovariance function $R(k)$ and the power spectral density $S_Z(\omega)$ form a Fourier transform pair:

$$S_Z(\omega) = \sum_{k=-\infty}^{\infty} R(k) e^{-i\omega k}$$

This gives a frequency-domain view of how power is distributed in the noise process.

3. Diagonalization via Fourier Basis As $n \to \infty$, Toeplitz matrices can be approximated by circulant matrices, which are diagonalized by the Discrete Fourier Transform (DFT) matrix F:

$$\mathcal{K}_n \approx F \Lambda F^\dagger$$

where Λ is diagonal and its entries approximate values of $S_Z(\omega)$.

4. Motivation This diagonalization "whitens" the noise in the frequency domain, transforming the channel into a set of independent parallel AWGN channels, each with different noise variances.

5. Water-Filling and Capacity This leads to the famous water-filling expression for capacity under a power constraint:

$$C = \frac{1}{2\pi} \int_{-\pi}^{\pi} \max\left(0, \frac{1}{2} \log\left(\frac{P(\omega)}{S_Z(\omega)}\right)\right) d\omega$$

where $P(\omega)$ is the power allocated at frequency ω, such that

$$\frac{1}{2\pi} \int_{-\pi}^{\pi} P(\omega)\, d\omega \leq P_{\text{total}}.$$

(20250408#306)

Give a conceptual overview of rate distortion theory:

- Consider a compression scheme where we allow a small probability of error:

$$\mathbb{P}\left(X^n \neq \hat{X}^n\right) \leq \epsilon, \quad \text{for some } \epsilon \in (0,1)$$

- This means that the decoder reconstructs the source sequence correctly with high probability:

$$\mathbb{P}\left(X^n = \hat{X}^n\right) \geq 1 - \epsilon$$

- The philosophy here is: when the reconstruction is correct, it is exactly correct (lossless), and this happens with high probability.
- More generally, in lossy compression with distortion, we relax exact correctness to approximate correctness.
  - That is, we allow distortion, but ensure it remains small with high probability.
- Thus, the scheme achieves:

$$\text{Probably approximately correct reconstrucion with high probability}$$

- In this sense, compression with a fidelity criterion (e.g., distortion or error probability) ensures that:

$$\text{The reconstruction is approximately correct, with high confidence}$$

(20250408#307)

Explain Lloyd-max quantization algorithm:

Why quantize signals? Because in many applications (e.g., audio, images, sensors), source signals are real-valued and continuous. To store or transmit these signals digitally, we must first discretize them — this is the role of quantization.

- The Lloyd-Max algorithm is a quantization method used to represent a signal using L levels.
- Let the source be a real-valued random variable:

$$X \in \mathcal{S} \subset \mathbb{R}$$

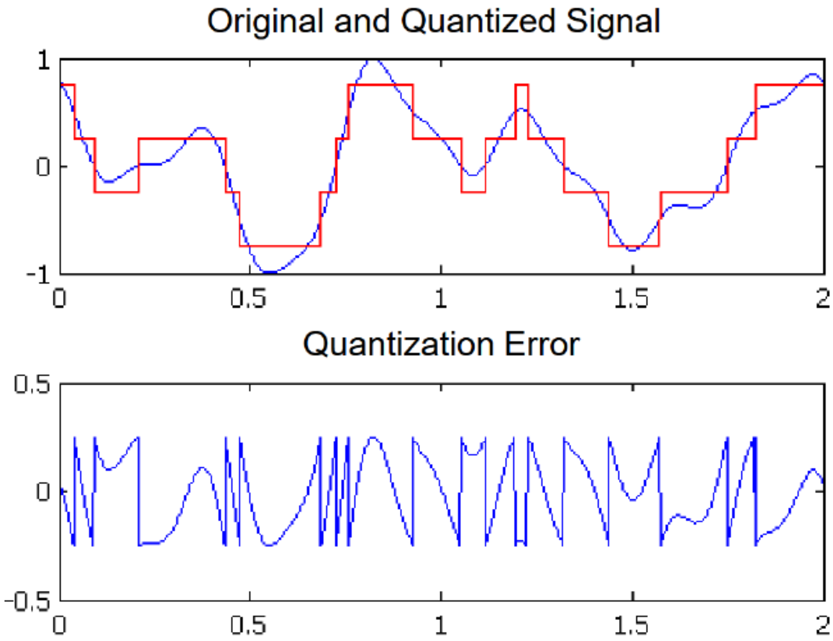
- The goal is to approximate X using a quantized value $\hat{X} \in \{c_1, \ldots, c_L\}$, where each c_i is a representative point (quantization level).

- Reconstruction incurs distortion. Typically, the distortion measure used is squared error:

  $$d(x, \hat{x}) = (x - \hat{x})^2$$

  The goal is to minimize the expected distortion:

  $$\mathbb{E}[(X - \hat{X})^2]$$



- The algorithm alternates between two steps:
  (a) **Nearest Neighbor Partitioning:**
  Given a set of quantization levels $\{c_1, \ldots, c_L\}$, partition the real line into intervals I_i such that each $x \in \mathcal{S}$ is assigned to the nearest level:

  $$x \in I_i \iff |x - c_i| \leq |x - c_j| \quad \forall j \neq i$$

  This defines the quantization intervals.
  (b) **Centroid Update:**
  Given the intervals I_i, update each level c_i to the centroid (conditional expectation) of X in the interval:

  $$c_i = \mathbb{E}[X \mid X \in I_i] = \frac{\int_{I_i} x f_X(x)dx}{\int_{I_i} f_X(x)dx}$$

  This minimizes the mean squared error within each partition.
  (c) Repeat steps (a) and (b) until convergence or a maximum number of iterations is reached.

- Each step locally minimizes distortion:
  (i) Step (a) assigns each input to the nearest level — locally optimal quantization.

(ii) Step (b) chooses the best representative (centroid) for each interval — locally optimal reconstruction.

- This is the classical **Lloyd-Max algorithm**, historically known as the precursor to the modern **k-means clustering algorithm**.

(20250408#308)

Give an example for Lloyd-Max algorithm for a model problem:

Goal: Quantize a scalar Gaussian random variable $X \sim \mathcal{N}(0,1)$ using a 2-level quantizer, and minimize the expected distortion under the squared error distortion measure.

Step 1: Initial Setup

- Let the number of quantization levels be $L = 2$.
- Assume reconstruction points are r_1 and r_2.
- Let the decision threshold be t, with:

$$\text{Quantization rule:} \quad \hat{X} = \begin{cases} r_1 & \text{if } X < t \\ r_2 & \text{if } X \geq t \end{cases}$$

- Squared error distortion: $d(x, \hat{x}) = (x - \hat{x})^2$

Step 2: Iterative Lloyd-Max Algorithm

(a) **Fix decision threshold t, compute centroids (reconstruction points):**

$$r_1 = \frac{\int_{-\infty}^{t} x f_X(x)\, dx}{\int_{-\infty}^{t} f_X(x)\, dx}, \quad r_2 = \frac{\int_{t}^{\infty} x f_X(x)\, dx}{\int_{t}^{\infty} f_X(x)\, dx}$$

where $f_X(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}$ is the PDF of $\mathcal{N}(0,1)$.

(b) **Fix r_1, r_2, update threshold t:**

$$t = \frac{r_1 + r_2}{2}$$

(c) **Repeat steps (a) and (b) until convergence:** Stop when $|r_i^{(k+1)} - r_i^{(k)}|$ and $|t^{(k+1)} - t^{(k)}|$ are below a desired threshold.

Step 3: Result (after convergence)

- After a few iterations, the algorithm converges to:

$$r_1 \approx -0.798, \quad r_2 \approx 0.798, \quad t = 0$$

- This means the input is quantized as:

$$\hat{X} = \begin{cases} -0.798 & \text{if } X < 0 \\ 0.798 & \text{if } X \geq 0 \end{cases}$$

- This solution minimizes the mean squared distortion over all 2-level quantizers for a standard Gaussian source.

Remarks:

- This algorithm is also known as the *K-means algorithm* for scalar quantization.
- Each step minimizes distortion locally:
  - Given thresholds, centroids are optimal (mean of distribution within the interval).
  - Given centroids, thresholds are optimal (midpoints).

(20250408#309)

State the goal of rarte distortion theory and give definition:

- **Goal:** To characterize the fundamental trade-off between *rate* (bits per symbol) and *distortion* (fidelity of reconstruction) in lossy compression schemes.
- **Setup:** Assume we are interested in asymptotic behavior as $n \to \infty$ for block length n.

  1. Distortion Measure:

  - Let A be a finite source alphabet, $|A| < \infty$.
  - Define a per-letter distortion function:

$$d : A \times A \to \mathbb{R}_+, \quad \text{with} \quad d_{\max} := \max_{a,a' \in A} d(a, a') < \infty$$

  - Interpreted as the distortion incurred when source symbol $x \in A$ is reconstructed as $\hat{x} \in A$.

  2. Block Distortion:

  - For sequences $x^n, \hat{x}^n \in A^n$, define the average distortion as:

$$d_n(x^n, \hat{x}^n) = \frac{1}{n} \sum_{i=1}^{n} d(x_i, \hat{x}_i)$$

3. (n, M) Code:

- Encoder: $f_n : A^n \to \{1, 2, \ldots, M\}$
- Decoder: $\phi_n : \{1, 2, \ldots, M\} \to A^n$
- The encoder and decoder together form a lossy compression scheme.

4. Code Performance Metrics:

- **Rate:** $R = \frac{\log M}{n}$ bits/symbol
- **Expected Distortion:**

$$\Delta_n = \mathbb{E}\left[d_n\left(X^n, \phi_n(f_n(X^n))\right)\right]$$

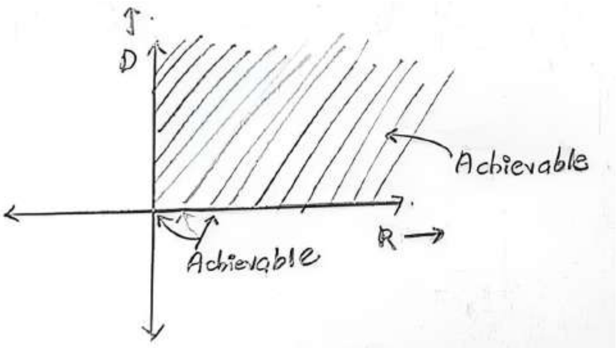
  where $X^n = (X_1, \ldots, X_n)$ is generated i.i.d. from the source distribution P_X.

5. Achievability:

- A pair (R, D) is said to be *achievable* if for all $\eta > 0$ and $\tau > 0$, there exists an (n, M_n) code such that:

$$\frac{\log M_n}{n} \leq R + \eta \quad \text{and} \quad \Delta_n \leq D + \tau$$

  The reason why we have $R + \eta$ and not $R - \eta$ is because we're looking at the asymptotic limit from above, not below.



6. Rate-Distortion Region and Function:

- The rate-distortion region is:

$$\mathcal{R} = \{(R, D) : (R, D) \text{ is achievable}\}$$

- The **Rate-Distortion Function** is defined as:

$$R(D) = \inf\{R : (R, D) \text{ is achievable}\}$$

- The **Distortion-Rate Function** is:

$$D(R) = \inf\{D : (R, D) \text{ is achievable}\}$$

(20250408#310)

Give some properties of rate-distortion region:

(a) **Monotonicity:** Suppose (R, D) is achievable. Then, for any (R', D') such that $R' \geq R$ and $D' \geq D$, the pair (R', D') is also achievable.

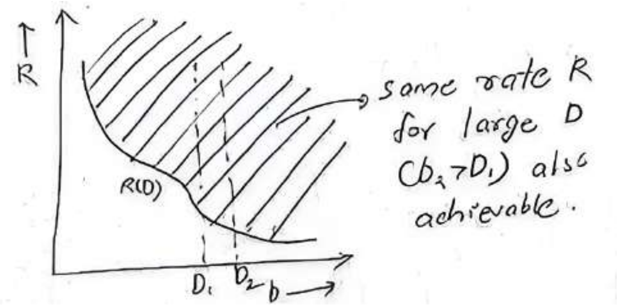
Argument: If there exists a code with parameters (n, M_n) such that:

$$\frac{\log M_n}{n} \leq R + \eta, \quad \Delta_n \leq D + \tau,$$

then clearly,

$$\frac{\log M_n}{n} \leq R' + \eta, \quad \Delta_n \leq D' + \tau,$$

since $R' \geq R$ and $D' \geq D$. Hence, (R', D') is also achievable.



(b) **$\mathcal{R}$ is closed:**

Suppose $(R_n, D_n) \in \mathcal{R}$ and $(R_n, D_n) \to (R, D)$. We want to show that $(R, D) \in \mathcal{R}$.

Argument: For a fixed $\eta > 0$ and $\tau > 0$, there exists a large enough n^* such that:

$$|R_{n^*} - R| \leq \eta/2, \quad |D_{n^*} - D| \leq \tau/2.$$

Since (R_{n^*}, D_{n^*}) is achievable, there exists a code (n^*, M_{n^*}) satisfying:

$$\frac{\log M_{n^*}}{n^*} \leq R + \eta, \quad \Delta_{n^*} \leq D + \tau.$$

Therefore, (R, D) is achievable, and so $\mathcal{R}$ is closed.

(c) **The function $D \mapsto R(D)$ is non-increasing:**

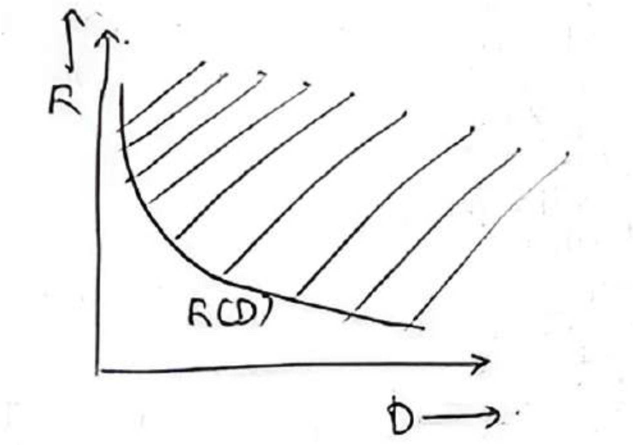
Argument: If $D_1 \leq D_2$, then achieving distortion D_1 requires at least as much rate as distortion D_2. Therefore:

$$R(D_1) \geq R(D_2),$$

which proves that $R(D)$ is non-increasing.

(d) **The function $D \mapsto R(D)$ is convex:**

Hint for proof: Convexity of the rate-distortion function comes from the convexity of the rate-distortion region $\mathcal{R}$.



If (R_1, D_1) and (R_2, D_2) are achievable, then for any $\lambda \in [0,1]$, the convex combination:

$$(R, D) = (\lambda R_1 + (1-\lambda)R_2,\ \lambda D_1 + (1-\lambda)D_2)$$

is also achievable via **time-sharing**. Hence, the function $R(D)$ is convex as it is the lower boundary of a convex set.

(20250408#311)

Prove the convexity of achievable rate distortion region:

Suppose (R_1, D_1) and (R_2, D_2) are both achievable. We want to prove that for any $\lambda \in [0,1]$, the convex combination

$$(R, D) = (\lambda R_1 + (1-\lambda)R_2,\ \lambda D_1 + (1-\lambda)D_2)$$

is also achievable.

Step 1: Use achievability of (R_1, D_1) and (R_2, D_2)

Let $\eta, \tau > 0$ be given arbitrarily. Since (R_1, D_1) and (R_2, D_2) are achievable, there exist codes of blocklength n such that:

$$\frac{\log M_n^{(1)}}{n} \le R_1 + \eta, \quad \Delta_n^{(1)} \le D_1 + \tau,$$

$$\frac{\log M_n^{(2)}}{n} \le R_2 + \eta, \quad \Delta_n^{(2)} \le D_2 + \tau.$$

Step 2: Construct a composite code by time-sharing

Let n be large, and set $n_1 = \lfloor \lambda n \rfloor$, $n_2 = n - n_1$. Apply the (R_1, D_1) code to the first n_1 symbols, and the (R_2, D_2) code to the remaining n_2 symbols.

Define the composite encoder f_n and decoder ϕ_n by combining $f_{n_1}^{(1)}$, $\phi_{n_1}^{(1)}$ and $f_{n_2}^{(2)}$, $\phi_{n_2}^{(2)}$ in a concatenated fashion.

Step 3: Analyze the rate

The total number of codewords is:

$$M_n = M_{n_1}^{(1)} \cdot M_{n_2}^{(2)} \Rightarrow \log M_n = \log M_{n_1}^{(1)} + \log M_{n_2}^{(2)}.$$

Therefore,

$$\frac{\log M_n}{n} = \frac{n_1}{n} \cdot \frac{\log M_{n_1}^{(1)}}{n_1} + \frac{n_2}{n} \cdot \frac{\log M_{n_2}^{(2)}}{n_2}.$$

Using the bounds:

$$\frac{\log M_n}{n} \leq \frac{n_1}{n}(R_1 + \eta) + \frac{n_2}{n}(R_2 + \eta) = \lambda R_1 + (1 - \lambda) R_2 + \eta.$$

Step 4: Analyze the distortion

The average distortion is:

$$\Delta_n = \frac{1}{n} \left(\sum_{i=1}^{n_1} \mathbb{E}[d(X_i, \hat{X}_i^{(1)})] + \sum_{i=n_1+1}^{n} \mathbb{E}[d(X_i, \hat{X}_i^{(2)})] \right).$$

So,

$$\Delta_n \leq \frac{n_1}{n}(D_1 + \tau) + \frac{n_2}{n}(D_2 + \tau) = \lambda D_1 + (1 - \lambda) D_2 + \tau.$$

Step 5: Conclusion

Since $\eta, \tau > 0$ were arbitrary, we conclude that (R, D) is achievable. Therefore, the rate-distortion region $\mathcal{R}$ is convex:

$$\lambda(R_1, D_1) + (1 - \lambda)(R_2, D_2) \in \mathcal{R}, \quad \forall \lambda \in [0, 1].$$

(20250408#312)

State Shannon's 1959 theorem on rate distortion function:

Let $(X_n)_{n\geq 1}$ be an i.i.d. sequence with marginal distribution P_X. Then, the rate-distortion function is given by:

$$R(D) = \inf_{P_{Y|X}:\mathbb{E}[d(X,Y)]\leq D} I(X;Y)$$

That is, the minimum rate required to encode the source with expected distortion no more than D is the minimum mutual information between X and Y, taken over all conditional distributions $P_{Y|X}$ satisfying the distortion constraint.

Interpretation:

This is in contrast to the channel coding problem. There:

- The channel $P_{Y|X}$ is *fixed* (i.e., nature determines how inputs are mapped to outputs).
- The goal is to *maximize* over input distributions P_X to determine capacity:

$$C = \sup_{P_X} I(X;Y)$$

In the rate-distortion setting:

- The source distribution P_X is *given* (i.e., the source produces data according to P_X).
- The goal is to *minimize* over reconstructions $P_{Y|X}$ that satisfy the distortion constraint, to determine the minimum rate.

(20250408#313)

For the example of binary source with hamming distortion, obtain rate-distortion function:

Let $X \sim \text{Bern}(p)$, with $\mathcal{A} = \{0, 1\}$ and Hamming distortion:

$$d(x, y) = 1\{x \neq y\}$$

We assume $p > \frac{1}{2}$ without loss of generality. The goal is to compute the rate-distortion function $R(D)$.

Step 1: Lower Bound on $R(D)$

The general rate-distortion lower bound is:

$$R(D) = \inf_{P_{Y|X}:\mathbb{E}[d(X,Y)]\leq D} I(X;Y)$$

We note:

$$I(X;Y) = H(X) - H(X|Y) = H(X) - H(X \oplus Y|Y)$$

Since $X \oplus Y$ is a function of X and Y, and Y is known, we have:

$$H(X \oplus Y|Y) \leq H(X \oplus Y) \Rightarrow I(X;Y) \geq H(X) - H(X \oplus Y)$$

as first inequality follows from conditioning reducing uncertainty.

Let $Z = X \oplus Y$ denote the error indicator (because Z is non-zero only when there is a bit mismatch between X and Y), which is Bernoulli with parameter $D = \mathbb{P}(X \neq Y) = \mathbb{P}(Z = 1)$. Then:

$$I(X;Y) \geq H(X) - H(Z) = H(p) - H(D)$$

Hence, the lower bound:

$$R(D) \geq H(p) - H(D)$$

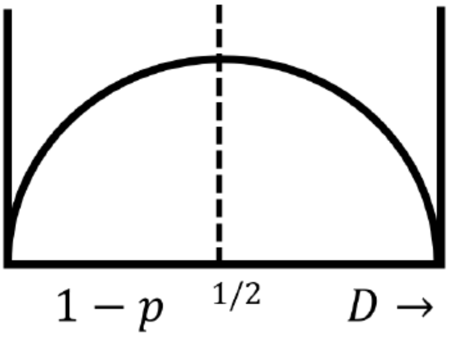
Case 1: $D \geq 1 - p$

Set $Y = 1$ deterministically (i.e., constant reproduction symbol). Then:

$$\mathbb{P}(X \neq Y) = \mathbb{P}(X = 0) = 1 - p \leq D$$

Thus, the distortion constraint is satisfied, and Y is independent of X. So $I(X;Y) = 0$, and:

$$R(D) = 0 \quad \text{for } D \geq 1 - p$$



Case 2: $D < 1 - p$

In this case, achieving zero rate is not possible. We seek to construct a conditional distribution $P_{Y|X}$ such that:

$$\mathbb{P}(X \neq Y) = D$$

Assume a symmetric test channel:

$$P(Y = X) = 1 - D, \quad P(Y \neq X) = D$$

Let $Z = X \oplus Y \sim \text{Bern}(D)$ be the error. If $Z \perp Y$, then:

$$I(X;Y) = H(X) - H(X \oplus Y) = H(p) - H(D)$$

Thus, the lower bound is tight:

$$R(D) = H(p) - H(D), \quad \text{for } D < 1 - p$$

Feasibility Condition:

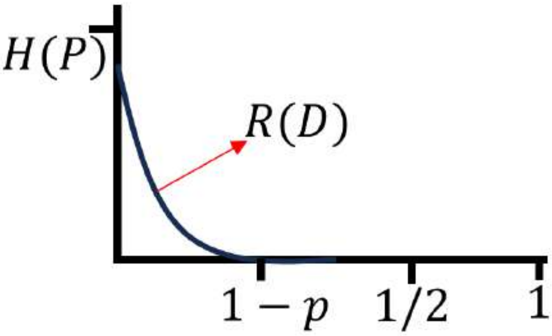
To ensure the marginal of X remains Bernoulli(p), define:

$$\mathbb{P}(Y = 1) = r, \quad \text{and solve:} \quad r(1 - D) + (1 - r)D = p \Rightarrow r = \frac{p - D}{1 - 2D}$$

We check the feasibility:

- $D < \frac{1}{2} \Rightarrow 1 - 2D > 0$
- $p > \frac{1}{2}, D < 1 - p \Rightarrow p - D > 0 \Rightarrow r > 0$
- $p < 1 \Rightarrow r < 1$

So, $0 < r < 1$, and the test channel is valid.



Conclusion:

$$R(D) = \begin{cases} 0 & \text{if } D \geq 1 - p \\ H(p) - H(D) & \text{if } D < 1 - p \end{cases}$$

(20250408#314)

Prove the converse for the rate-distortion theorem:

Suppose we have a sequence of (n, M_n) codes. Consider the standard coding setup:

$$X^n \xrightarrow{f_n} \{1, 2, \ldots, M_n\} \xrightarrow{\phi_n} Y^n$$

Since the decoder maps to at most M_n distinct output sequences, we have:

$$\log M_n \geq H(Y^n)$$

By the data processing inequality:

$$H(Y^n) \geq I(X^n; Y^n) = H(X^n) - H(X^n|Y^n)$$

Now, since $X^n \sim P_X^{\otimes n}$, the components X_i are i.i.d., and the same holds for the output components Y_i. Therefore,

$$I(X^n; Y^n) \geq \sum_{i=1}^{n} I(X_i; Y_i)$$

Now consider distortion. Let $d_i = \mathbb{E}[d(X_i, Y_i)]$. Since each (X_i, Y_i) pair induces a joint distribution over $\mathcal{X} \times \mathcal{Y}$, and $R(d)$ is defined as:

$$R(d) = \inf_{P_{Y|X}:\mathbb{E}[d(X,Y)]\leq d} I(X; Y)$$

we get:

$$I(X_i; Y_i) \geq R(d_i)$$

Hence,

$$\frac{1}{n} \log M_n \geq \frac{1}{n} \sum_{i=1}^{n} R(d_i) \geq R\left(\frac{1}{n} \sum_{i=1}^{n} d_i\right)$$

Let $D_n = \mathbb{E}[d_n(X^n, Y^n)] = \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}[d(X_i, Y_i)]$. Then:

$$\frac{1}{n} \log M_n \geq R(D_n)$$

If the code is good, meaning the average distortion $D_n \leq D + \tau$ for any $\tau > 0$, then:

$$\frac{1}{n} \log M_n \geq R(D + \tau)$$

Since $\tau > 0$ is arbitrary and $R(D)$ is non-increasing, we conclude:

$$\liminf_{n\to\infty} \frac{1}{n} \log M_n \geq R(D)$$

This completes the proof of the converse.

Claim: $R(D) = \inf_{P_{Y|X}:\mathbb{E}[d(X,Y)]\leq D} I(X; Y)$ is convex and non-increasing.

Proof:

- The function $R(D)$ is non-increasing since increasing the allowed distortion enlarges the feasible set of test channels $P_{Y|X}$, thereby potentially reducing mutual information.

- The function $R(D)$ is convex due to the convexity of mutual information in the conditional distribution $P_{Y|X}$, and because the constraint set:

  $$\left\{ P_{Y|X} : \sum_{x,y} P_X(x) P_{Y|X}(y|x) d(x,y) \leq D \right\}$$

  is convex. Hence, the infimum of a convex function over a convex set is convex.

(20250409#315)

Show this $S(D)$ is non-increasing and convex:

$$S(D) = \inf_{P_{Y|X}:\mathbb{E}[d(X,Y)]\leq D} I(X;Y),$$

Fix a source distribution P_X, and let the distortion measure be

$$d : \mathcal{A} \times \mathcal{A} \to \mathbb{R}_+,$$

where $\mathcal{A}$ is a finite alphabet. For each distortion threshold $D \geq 0$, define the function:

$$S(D) = \inf_{P_{Y|X}:\mathbb{E}[d(X,Y)]\leq D} I(X;Y),$$

where the infimum is over all conditional distributions $P_{Y|X}$ satisfying the expected distortion constraint under the fixed marginal P_X.

Claim: The function $D \mapsto S(D)$ is non-increasing and convex.

- **Why is $S(D)$ non-increasing?**

  If $D_2 > D_1$, then the constraint set

  $$\{P_{Y|X} : \mathbb{E}[d(X,Y)] \leq D_1\}$$

  is a subset of the corresponding set for D_2. Hence, the infimum over a larger feasible set (for higher D) cannot be larger:

  $$S(D_2) \leq S(D_1).$$

- **Why is $S(D)$ convex?**

  The set of admissible channels $P_{Y|X}$ for which $\mathbb{E}[d(X,Y)] \leq D$ and $\sum_b P_{Y|X}(b|a) = 1\ \forall a \in \mathcal{A}$, $P_{Y|X}(b|a) \geq 0$, is a convex polytope.

  The dimensionality of the conditional distribution $P_{Y|X}$ is $|\mathcal{A}|(|\mathcal{A}|-1)$ since for each a, the conditional probabilities over $b \in \mathcal{A}$ must sum to 1.

  Each constraint (non-negativity, normalization, distortion bound) is linear in $P_{Y|X}(b|a)$, so the feasible set is convex.

  To show $S(D)$ is convex: for any $D_1, D_2 \geq 0$, let

  $$P^{(1)}_{Y|X} \text{ and } P^{(2)}_{Y|X}$$

  achieve (or nearly achieve) $S(D_1)$ and $S(D_2)$ respectively. Consider the convex combination

  $$P^{(\lambda)}_{Y|X} = \lambda P^{(1)}_{Y|X} + (1-\lambda)P^{(2)}_{Y|X}, \quad \lambda \in [0,1].$$

  The distortion under this convex combination satisfies:

  $$\mathbb{E}[d(X,Y)] \leq \lambda D_1 + (1-\lambda)D_2,$$

so $P^{(\lambda)}_{Y|X} \in \mathcal{C}(\lambda D_1 + (1-\lambda)D_2)$, the feasible set at average distortion.
The mutual information is convex in $P_{Y|X}$ for fixed P_X, so:

$$I_{P^{(\lambda)}_{Y|X}}(X;Y) \leq \lambda I_{P^{(1)}_{Y|X}}(X;Y) + (1-\lambda) I_{P^{(2)}_{Y|X}}(X;Y),$$

which implies:

$$S(\lambda D_1 + (1-\lambda)D_2) \leq \lambda S(D_1) + (1-\lambda)S(D_2).$$

Hence, $S(D)$ is convex and non-increasing.

(20250409#316)

Show the achievability of $R(D) \leq S(D)$:

Objective: Prove the achievability part of the rate-distortion theorem, i.e.,

$$R(D) \leq S(D),$$

where $S(D) = \inf_{P_{Y|X}:\mathbb{E}[d(X,Y)]\leq D} I(X;Y)$.

We want to show that for any distortion level D, there exists a code with rate close to $I(X;Y)$ and expected distortion not exceeding D.

Step 1: Fix a test channel $P_{Y|X}$

- Choose a conditional distribution $P_{Y|X}$ such that $\mathbb{E}[d(X,Y)] \leq D$, and $I(X;Y)$ is finite.
- Let $P_{XY} = P_X P_{Y|X}$, and P_Y be the marginal of Y.

Step 2: Random Codebook Generation

- Fix blocklength n and number of codewords M_n.
- Generate a random codebook:

$$\mathcal{C} = \{y^n(1), y^n(2), \ldots, y^n(M_n)\}, \quad y^n(i) \sim P_Y^n \text{ i.i.d.}$$

Step 3: Typical Set Definition

Let $\tau > 0$ be a small constant. Define the δ-typical set $A(n, \tau) \subseteq \mathcal{X}^n \times \mathcal{Y}^n$ as:

$$\begin{aligned} A(n,\tau) = \Big\{ (x^n, y^n) \in \mathcal{X}^n \times \mathcal{Y}^n : \ & \left| -\frac{1}{n} \log P_{X^n}(x^n) - H(X) \right| < \tau, \\ & \left| -\frac{1}{n} \log P_{Y^n}(y^n) - H(Y) \right| < \tau, \\ & \left| -\frac{1}{n} \log P_{X^nY^n}(x^n, y^n) - H(X,Y) \right| < \tau, \\ & |d(x^n, y^n) - \mathbb{E}[d(X,Y)]| < \tau \Big\}. \end{aligned}$$

Step 4: Encoding Rule

- Given a source sequence x^n, find the smallest index $w \in \{1, \ldots, M_n\}$ such that:
$$(x^n, y^n(w)) \in A(n, \tau).$$
- If no such w exists, declare an encoding failure and output a default index (say, M_n).
- Define the encoder as $f_n(x^n) = w$.

Step 5: Decoding Rule

Given index w, output the codeword $y^n(w)$ as the reproduction sequence:

$$\hat{x}^n = y^n(w).$$

Step 6: Error Analysis

The error occurs when no codeword in the codebook is jointly typical with x^n:

$$P_e^{(n)} = \mathbb{P}\left[\forall w, (x^n, y^n(w)) \notin A(n,\tau)\right].$$

We evaluate:

$$\begin{aligned} P_e^{(n)} &= \sum_{x^n} P_{X^n}(x^n) \mathbb{P}\left[\forall w, (x^n, y^n(w)) \notin A(n,\tau)\right] \\ &= \sum_{x^n} P_{X^n}(x^n) \left(1 - \mathbb{P}_{Y^n}\left[(x^n, Y^n) \in A(n,\tau)\right]\right)^{M_n}. \end{aligned}$$

Because the codewords $y^n(w) \sim P_Y^n$ i.i.d., and are independent of x^n, we can use the inequality:

$$(1-u)^M \leq (1-v) + e^{-Mu} \quad \text{for } u \leq v, u, v \in [0,1].$$

Let $\delta_n := \min_{x^n} \mathbb{P}_{Y^n}\left[(x^n, Y^n) \in A(n,\tau)\right]$. Then:

$$P_e^{(n)} \leq 1 - P_{X^nY^n}(A(n,\tau)) + e^{-M_n \cdot 2^{-n(I(X;Y)+3\tau)}}.$$

Step 7: Rate Constraint for Vanishing Error

To ensure the error probability vanishes as $n \to \infty$, it suffices to choose:

$$M_n \geq 2^{n(I(X;Y)+4\tau)}.$$

Thus, the rate $R_n = \frac{1}{n} \log M_n$ satisfies:

$$R_n \leq I(X;Y) + 4\tau.$$

Conclusion:

Since τ is arbitrary and can be made arbitrarily small, we conclude:

$$R(D) \leq I(X;Y) \quad \text{for any } P_{Y|X} \text{ such that } \mathbb{E}[d(X,Y)] \leq D.$$

Taking the infimum over all such channels completes the achievability proof:

$$R(D) \leq \inf_{P_{Y|X}:\mathbb{E}[d(X,Y)]\leq D} I(X;Y) = S(D).$$

(20250409#317)

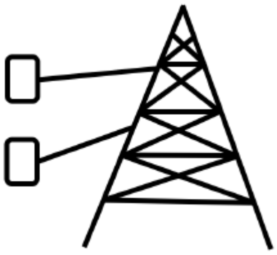
Explain multiple access channels, broadcast channels, multicast channels, polar codes and back from infinity analysis:

1. Multiple Access Channels (MAC) – Uplink Scenario

Consider a simple two-user Gaussian Multiple Access Channel (MAC) where the received signal is:

$$Y = X_1 + X_2 + Z,$$

where X_1, X_2 are the transmitted signals of users 1 and 2 respectively, and $Z \sim \mathcal{N}(0,1)$ is AWGN (additive white Gaussian noise).



- Suppose power constraints are $\mathbb{E}[X_i^2] \leq P_i$ for $i = 1, 2$.

- The capacity region is the set of all rate pairs (R_1, R_2) satisfying:

$$\begin{aligned} R_1 &\leq \frac{1}{2}\log(1 + P_1), \\ R_2 &\leq \frac{1}{2}\log(1 + P_2), \\ R_1 + R_2 &\leq \frac{1}{2}\log(1 + P_1 + P_2). \end{aligned}$$

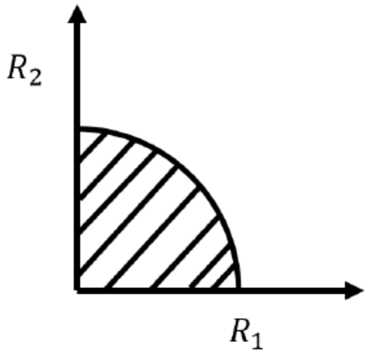
- The third constraint dominates: the total rate is bounded by the capacity of the sum-power channel.
- If user 1 is transmitting at maximum allowable rate $R_1 = \frac{1}{2}\log(1 + P_1)$, then user 2 is limited to:
$$R_2 \leq \frac{1}{2}\log\left(\frac{1 + P_1 + P_2}{1 + P_1}\right) = \frac{1}{2}\log\left(1 + \frac{P_2}{1 + P_1}\right).$$

Asymptotic approximation for high P_1, P_2:

$$\frac{1}{2}\log(1 + P_1 + P_2) \approx \frac{1}{2}\log(P_2) + \log(P_1),$$

when $P_1 \gg 1, P_2 \gg 1$, showing the joint benefit of power pooling.

2. Broadcast Channels (BC) – Downlink Scenario



- The capacity region for general broadcast channels is not fully known.
- For Gaussian BCs (e.g., superposition coding), capacity regions are known and can be achieved using successive decoding.
- For arbitrary channels, determining capacity is an open problem in information theory.

3. Multicast

This refers to sending the same data to multiple users over the network:

- All users must decode the same message.
- Performance depends on the user with the weakest channel (bottleneck).

- Applications include content delivery networks, video streaming, etc.

4. Polar Codes

Introduced by Arıkan, polar codes are the first family of capacity-achieving codes with low encoding and decoding complexity.

- Based on the concept of *channel polarization* where synthetic channels become either nearly perfect or completely noisy.
- Constructed via recursive application of a transform (e.g., $G_N = B_N F^{\otimes n}$, where $F = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$).
- As $n \to \infty$, a fraction $\approx I(W)$ of the channels are nearly noiseless (good), and the rest are useless (bad).
- Complexity: $O(n \log n)$ for both encoding and decoding.

5. Finite Blocklength (Back from Infinity Analysis)

In practical systems, blocklength n is finite and we cannot rely on $n \to \infty$ asymptotics. Finite-blocklength analysis provides tight performance bounds.

- For any coding scheme, the probability of decoding failure can be bounded using concentration inequalities.
- A general form of the bound:

  $$\Pr\left[I(X;Y) \in A(n,\tau)\right] + (M_n)^{-nJ(X,Y)+\eta},$$

  where $J(X,Y)$ measures rate-redundancy, and η accounts for slackness.
- Another bound:

  $$\Pr\left[\log \frac{1}{P_{Y^n|X^n}(Y^n|X^n)} \geq n\right] \leq (M_n - 1)^{-n},$$

  used in strong converse and error exponent analysis.
- Finite blocklength capacity approximations involve normal approximation:

  $$R(n,\varepsilon) \approx C - \sqrt{\frac{V}{n}} Q^{-1}(\varepsilon),$$

  where C is channel capacity, V is the channel dispersion, and Q^{-1} is the inverse of the Gaussian tail function.

(Assignment-1.#318)

1. Exercises on Sequences

- Recall the notation that $\leq_n$ stands for the relation "is less than or equal for all sufficiently large n". Recall also the definition of the *limsup* of a sequence:

$$\limsup_{n\to\infty} x_n = \inf_{n>0} \sup_{m\geq n} x_m.$$

- Suppose that for each $\varepsilon > 0$, we have $a_n < a + \varepsilon$. Show that

$$\limsup_{n\to\infty} a_n \leq a.$$

- Let $a_n \leq b_n$. Show that

$$\limsup_{n\to\infty} a_n \leq \limsup_{n\to\infty} b_n.$$

- Let $a = \limsup_{n\to\infty} a_n \in \mathbb{R}$. Show that for every $\varepsilon > 0$, the inequality

$$a_n > a - \varepsilon$$

  occurs infinitely often.

- What are the analogous statements for lim inf?

- Show that

$$\liminf_{n\to\infty} a_n \leq \limsup_{n\to\infty} a_n$$

- Show that

$$\liminf_{n\to\infty} a_n = \limsup_{n\to\infty} a_n = a \in \mathbb{R}$$

  if and only if the following holds: for every $\varepsilon > 0$, there exists an N such that $n \geq N$ implies

$$|a_n - a| \leq \varepsilon.$$

  This establishes that the usual notion of a limit and the one via lim sup and lim inf are equivalent.

---

1. Suppose that for each $\varepsilon > 0$, we have $a_n < a + \varepsilon$. Show that $\limsup_{n\to\infty} a_n \leq a$.

Solution:

By the definition,

$$\limsup_{n\to\infty} a_n = \inf_{n\geq 1} \sup_{m\geq n} a_m.$$

Since for every $\varepsilon > 0$, we have $a_n < a + \varepsilon$ for all n, then $\sup_{m\geq n} a_m < a + \varepsilon$ for all n. Hence,

$$\inf_{n} \sup_{m\geq n} a_m \leq a + \varepsilon \quad \text{for all } \varepsilon > 0.$$

Thus, $\limsup_{n\to\infty} a_n \le a + \varepsilon$ for all $\varepsilon > 0$. Taking $\varepsilon \to 0$, we conclude

$$\limsup_{n\to\infty} a_n \le a.$$

2. Let $a_n \le b_n$. Show that $\limsup_{n\to\infty} a_n \le \limsup_{n\to\infty} b_n$.

Solution:

For all $m \ge n$, we have $a_m \le b_m$, so $\sup_{m\ge n} a_m \le \sup_{m\ge n} b_m$ for all n. Therefore,

$$\inf_n \sup_{m\ge n} a_m \le \inf_n \sup_{m\ge n} b_m.$$

Hence,

$$\limsup_{n\to\infty} a_n \le \limsup_{n\to\infty} b_n.$$

3. Let $a = \limsup_{n\to\infty} a_n \in \mathbb{R}$. Show that for every $\varepsilon > 0$, the inequality $a_n > a - \varepsilon$ occurs infinitely often.

Solution:

Assume for contradiction that $a_n > a - \varepsilon$ occurs only finitely many times. Then there exists N such that for all $n \ge N$, $a_n \le a - \varepsilon$. Therefore,

$$\sup_{m\ge n} a_m \le a - \varepsilon \quad \text{for all } n \ge N,$$

implying

$$\limsup_{n\to\infty} a_n \le a - \varepsilon,$$

contradicting the fact that $\limsup_{n\to\infty} a_n = a$. Hence, $a_n > a - \varepsilon$ must occur infinitely often.

4. What are the analogous statements for $\liminf$?

Solution:

Analogously:

- If for every $\varepsilon > 0$, $a_n > a - \varepsilon$ eventually holds, then $\liminf_{n\to\infty} a_n \ge a$. - If $a_n \ge b_n$ for all n, then $\liminf a_n \ge \liminf b_n$. - If $a = \liminf a_n$, then for every $\varepsilon > 0$, the inequality $a_n < a + \varepsilon$ occurs infinitely often.

5. Show that $\liminf_{n\to\infty} a_n \le \limsup_{n\to\infty} a_n$.

Solution:

By definition:

$$\liminf a_n = \sup_n \inf_{m\geq n} a_m, \quad \limsup a_n = \inf_n \sup_{m\geq n} a_m.$$

For any fixed n,

$$\inf_{m\geq n} a_m \leq \sup_{m\geq n} a_m.$$

Taking the supremum of the left and infimum of the right over n, we get

$$\liminf a_n \leq \limsup a_n.$$

6. Show that $\liminf a_n = \limsup a_n = a \in \mathbb{R}$ if and only if for every $\varepsilon > 0$, there exists N such that $n \geq N$ implies $|a_n - a| < \varepsilon$.

Solution:

($\Rightarrow$) If $\liminf a_n = \limsup a_n = a$, then for every $\varepsilon > 0$ there exists N such that:

$$a - \varepsilon < \inf_{m\geq n} a_m \leq a_n \leq \sup_{m\geq n} a_m < a + \varepsilon \quad \text{for all } n \geq N,$$

hence $|a_n - a| < \varepsilon$.

($\Leftarrow$) If $|a_n - a| < \varepsilon$ for all $n \geq N$, then:

$$a - \varepsilon < a_n < a + \varepsilon \Rightarrow \liminf a_n \geq a - \varepsilon, \quad \limsup a_n \leq a + \varepsilon.$$

Taking $\varepsilon \to 0$, we get $\liminf a_n = \limsup a_n = a$.

(Assignment-1.2#319)

Markov's inequality and Chebyshev's inequality

(a) **(Markov's inequality)** For any nonnegative random variable X and any $t > 0$, show that

$$\Pr\{X \geq t\} \leq \frac{EX}{t}. \quad (3.31)$$

Exhibit a random variable that achieves this inequality with equality.

(b) **(Chebyshev's inequality)** Let Y be a random variable with mean μ and variance σ^2. By letting $X = (Y - \mu)^2$, show that

$$\Pr\{|Y - \mu| \geq \epsilon\} \leq \frac{\sigma^2}{\epsilon^2}. \quad (3.32)$$

for any $\epsilon > 0$.

(c) **(Weak law of large numbers)** Let $Z_1, Z_2, \ldots, Z_n$ be a sequence of i.i.d. random variables with mean μ and variance σ^2. Let $\bar{Z}_n = \frac{1}{n}\sum_{i=1}^{n} Z_i$ be the sample mean. Show that

$$\Pr\{|\bar{Z}_n - \mu| \geq \epsilon\} \leq \frac{\sigma^2}{n\epsilon^2}. \quad (3.33)$$

Thus, $\Pr\{|\bar{Z}_n - \mu| \geq \epsilon\} \to 0$ as $n \to \infty$. This is known as the weak law of large numbers.

(Assignment-1.3#320)

Piece of cake: A cake is sliced roughly in half, the largest piece being chosen each time, the other pieces discarded. We will assume that a random cut creates pieces of proportions

$$P = \begin{cases} \left(\frac{2}{3}, \frac{1}{3}\right) & \text{with probability } \frac{3}{4} \\ \left(\frac{3}{5}, \frac{2}{5}\right) & \text{with probability } \frac{1}{4} \end{cases}$$

Thus, for example, the first cut (and choice of largest piece) may result in a piece of size $\frac{3}{5}$. Cutting and choosing from this piece might reduce it to $\left(\frac{3}{5}\right)\left(\frac{2}{3}\right)$ at time 2, and so on. How large, to first order in the exponent, is the piece of cake after n cuts?

(Assignment-1.4#321)

AEP and source coding. A discrete memoryless source emits a sequence of statistically independent binary digits with probabilities $p(1) = 0.005$ and $p(0) = 0.995$. The digits are taken 100 at a time and a binary codeword is provided for every sequence of 100 digits containing three or fewer 1's.

(a) Assuming that all codewords are the same length, find the minimum length required to provide codewords for all sequences with three or fewer 1's.

(b) Calculate the probability of observing a source sequence for which no codeword has been assigned.

(c) Use Chebyshev's inequality to bound the probability of observing a source sequence for which no codeword has been assigned. Compare this bound with the actual probability computed in part (b).

(Assignment-1.5#322)

Calculation of typical set.

To clarify the notion of a typical set $A_\epsilon^{(n)}$ and the smallest set of high probability $B_\delta^{(n)}$, we will calculate the set for a simple example. Consider a sequence of i.i.d. binary random variables, $X_1, X_2, \ldots, X_n$, where the probability that $X_i = 1$ is 0.6 (and therefore the probability that $X_i = 0$ is 0.4).

(a) Calculate $H(X)$.

(b) With $n = 25$ and $\epsilon = 0.1$, which sequences fall in the typical set $A_\epsilon^{(n)}$? What is the probability of the typical set? How many elements are there in the typical set? (This involves computation of a table of probabilities for sequences with k 1's, $0 \leq k \leq 25$, and finding those sequences that are in the typical set.)

(c) How many elements are there in the smallest set that has probability 0.9?

(d) How many elements are there in the intersection of the sets in parts (b) and (c)? What is the probability of this intersection?

---

1. $H(X) = -0.6 \log 0.6 - 0.4 \log 0.4 = 0.97095$ bits.

2. By definition, $A_\epsilon^{(n)}$ for $\epsilon = 0.1$ is the set of sequences such that $-\frac{1}{n} \log p(x^n)$ lies in the range $(H(X) - \epsilon, H(X) + \epsilon)$, i.e., in the range $(0.87095, 1.07095)$.

   Examining the last column of the table, it is easy to see that the typical set is the set of all sequences with the number of ones lying between 11 and 19.

   The probability of the typical set can be calculated from the cumulative probability column. The probability that the number of 1's lies between 11 and 19 is equal to:

   $$F(19) - F(10) = 0.970638 - 0.034392 = 0.936246.$$

   Note that this is greater than $1 - \epsilon$, i.e., n is large enough for the probability of the typical set to be greater than $1 - \epsilon$.

   The number of elements in the typical set can be found using the third column:

   $$|A_\epsilon^{(n)}| = \sum_{k=11}^{19} \binom{n}{k} = \sum_{k=0}^{19} \binom{n}{k} - \sum_{k=0}^{10} \binom{n}{k} = 33486026 - 7119516 = 26366510.$$

   Note that the upper and lower bounds for the size of the $A_\epsilon^{(n)}$ can be calculated as

   $$2^{n(H+\epsilon)} = 2^{25(0.97095+0.1)} = 2^{26.77} \approx 1.147365 \times 10^8,$$

   and

   $$(1-\epsilon)2^{n(H-\epsilon)} = 0.9 \times 2^{25(0.97095-0.1)} = 0.9 \times 2^{21.9875} \approx 3742308.$$

   Both bounds are very loose.

3. To find the smallest set $B_\delta^{(n)}$ of probability 0.9, we imagine that we are filling a bag with pieces such that we want to reach a certain weight with the minimum number of pieces. To minimize the number of pieces that we use, we should use the largest possible pieces. In this case, it corresponds to using the sequences with the highest probability.

   We keep putting the high-probability sequences into this set until we reach a total probability of 0.9. Looking at the fourth column of the table, it is clear that the probability of a sequence increases monotonically with k. Thus the set consists of sequences with $k = 25, 24, \ldots$ until the total probability is 0.9.

   Using the cumulative probability column, it follows that the set $B_\delta^{(n)}$ consists of sequences with $k \geq 13$ and some sequences with $k = 12$. The sequences with $k \geq 13$ provide a total probability of $1 - 0.153768 = 0.846232$ to the set $B_\delta^{(n)}$. The remaining probability of $0.9 - 0.846232 = 0.053768$ should come from sequences with $k = 12$. The number of such sequences needed to fill this probability is at least:

$$\frac{0.053768}{p(x^n)} = \frac{0.053768}{1.460813 \times 10^{-8}} = 3680691.1,$$

   which we round up to 3680691.

   Thus the smallest set with probability 0.9 has $33554432 - 16777216 + 3680691 = 20457907$ sequences. Note that the set $B_\delta^{(n)}$ is not uniquely defined—it could include any 3680691 sequences with $k = 12$. However, the size of the smallest set is well defined.

4. The intersection of the sets $A_\epsilon^{(n)}$ and $B_\delta^{(n)}$ in parts (b) and (c) consists of all sequences with k between 13 and 19, and 3680691 sequences with $k = 12$.

   The probability of this intersection is:

$$0.970638 - 0.153768 + 0.053768 = 0.870638,$$

   and the size of this intersection is:

$$33486026 - 16777216 + 3680691 = 20389501.$$

(Assignment-2.1#323)

Consider a discrete memoryless source DMS on $\mathcal{A}$ with PMF p. Recall that $s_q(n,\epsilon)$ is the minimum q-weight of sets whose p-probability is at least $1-\epsilon$. Prove the following theorem which was discussed in class: For any ϵ satisfying $0<\epsilon<1$, we have

$$\lim_{n\to\infty}\frac{\log s_q(n,\epsilon)}{n}=-\sum_{x\in\mathcal{A}}p(x)\log\frac{p(x)}{q(x)}.$$

Let $P_n(x_1,x_2,\dots,x_n)=\prod_{i=1}^{n}p(x_i)$ Let $Q_n(x_1,x_2,\dots,x_n)=\prod_{i=1}^{n}q(x_i)$

Q-wt of set $C_n=\sum_{(x_1,x_2,\dots,x_n)\in C_n}Q_n(x_1,x_2,\dots,x_n)$

We want to find min Q-wt of sets with $P\geq 1-\epsilon$ Let B_n be the set of sequences with $P\geq 1-\epsilon$

$$P(B_n)\geq 1-\epsilon$$

$$\sum_{(x_1,\dots,x_n)\in B_n}P_n(x_1,\dots,x_n)\geq 1-\epsilon$$

Let us define $A(n,\delta)$ to be a typical set as:

$$A(n,\delta)=\left\{(x_1,x_2,\dots,x_n)\in\mathcal{A}^n:\left|\frac{1}{n}\log\frac{P_n(x_1,\dots,x_n)}{Q_n(x_1,\dots,x_n)}-D(P||Q)\right|<\delta\right\}$$

where $\frac{1}{n}\log\frac{P_n(x_1,\dots,x_n)}{Q_n(x_1,\dots,x_n)}=\frac{1}{n}\sum_{i=1}^{n}\log\frac{p(x_i)}{q(x_i)}$. We know that $P(A(n,\delta))\geq 1-\epsilon$ for sufficiently large n.

Arrange sequences in B_n in decreasing order of $\frac{P_n(\cdot)}{Q_n(\cdot)}$ (Because we want to find a set with high prob. and low Q-wt).

We have $\sum_{(x_1,\dots,x_n)\in B_n}\frac{P_n(x_1,\dots,x_n)}{Q_n(x_1,\dots,x_n)}Q_n(x_1,\dots,x_n)\geq 1-\epsilon$.

So, now B_n is a contender of min Q-wt set with $P\geq 1-\epsilon$.

We know about the properties of the typical set, so we look for $B_n\cap A(n,\delta)$.

$$\begin{aligned}P(B_n\cap A(n,\delta))&=P(B_n)+P(A(n,\delta))-P(B_n\cup A(n,\delta))\\&\geq(1-\epsilon)+(1-\epsilon)-1\\&=1-2\epsilon\end{aligned}$$

From the previous steps, we have:

$$\sum_{(x_1,\dots,x_n)\in B_n\cap A(n,\delta)}\frac{P_n(x_1,\dots,x_n)}{Q_n(x_1,\dots,x_n)}\geq 1-2\epsilon\quad\cdots(1)$$

So, we are running short of almost ϵ probability.

We pick some min Q-wt seq. from $A(n,\delta)^c$ with prob $\leq \epsilon$.

$$\sum_{(x_1,\ldots,x_n)\in B_n\setminus(B_n\cap A(n,\delta))} \frac{P_n(x_1,\ldots,x_n)}{Q_n(x_1,\ldots,x_n)} \leq \epsilon \quad \cdots(2)$$

Adding (1) and (2):

$$\sum_{(x_1,\ldots,x_n)\in B_n} \frac{P_n(x_1,\ldots,x_n)}{Q_n(x_1,\ldots,x_n)} \geq 1-2\epsilon + \sum_{(x_1,\ldots,x_n)\in B_n\setminus(B_n\cap A(n,\delta))} \frac{P_n(\cdot)}{Q_n(\cdot)} - \epsilon$$

This doesn't look right. Let's restart from the image.

From (1):

$$\sum_{(x_1,\ldots,x_n)\in B_n\cap A(n,\delta)} \frac{P_n(x_1,\ldots,x_n)}{Q_n(x_1,\ldots,x_n)} \geq 1-2\epsilon \quad \cdots(1)$$

We pick some min Q-wt seq. from $A(n,\delta)^c$ with prob $\leq \epsilon$. Let the set of these sequences be $B_n \setminus (B_n \cap A(n,\delta))$.

$$\sum_{(x_1,\ldots,x_n)\in B_n\setminus(B_n\cap A(n,\delta))} \frac{P_n(x_1,\ldots,x_n)}{Q_n(x_1,\ldots,x_n)} \leq \epsilon \quad \cdots(2)$$

Adding (1) and (2) doesn't seem to lead anywhere directly.

We know that bounds on prob of typical seq is $2^{-n(H(X)+\delta)} \leq P_n(x_1,\ldots,x_n) \leq 2^{-n(H(X)-\delta)}$. For our typical set,

$$2^{-n(D(P||Q)+\delta)} \leq \frac{Q_n(\cdot)}{P_n(\cdot)} \leq 2^{-n(D(P||Q)-\delta)}$$

This implies

$$2^{n(D(P||Q)-\delta)} \leq \frac{P_n(\cdot)}{Q_n(\cdot)} \leq 2^{n(D(P||Q)+\delta)}$$

Putting in (1):

$$\sum_{(x_1,\ldots,x_n)\in B_n\cap A(n,\delta)} 2^{n(D(P||Q)-\delta)} Q_n(x_1,\ldots,x_n) \leq \sum_{(x_1,\ldots,x_n)\in B_n\cap A(n,\delta)} \frac{P_n(\cdot)}{Q_n(\cdot)} Q_n(\cdot) \geq 1-2\epsilon$$

$$2^{n(D(P||Q)-\delta)} \sum_{(x_1,\ldots,x_n)\in B_n\cap A(n,\delta)} Q_n(x_1,\ldots,x_n) \geq 1-2\epsilon$$

$$\sum_{(x_1,\ldots,x_n)\in B_n\cap A(n,\delta)} Q_n(x_1,\ldots,x_n) \geq (1-2\epsilon)2^{-n(D(P||Q)-\delta)}$$

Consider the Q-weight of B_n:

$$\begin{aligned}\sum_{(x_1,\ldots,x_n)\in B_n} Q_n(x_1,\ldots,x_n) &= \sum_{(x_1,\ldots,x_n)\in B_n\cap A(n,\delta)} Q_n(x_1,\ldots,x_n) + \sum_{(x_1,\ldots,x_n)\in B_n\setminus(B_n\cap A(n,\delta))} Q_n(x_1,\ldots,x_n) \\ &\geq (1-2\epsilon)2^{-n(D(P||Q)-\delta)} + \sum_{(x_1,\ldots,x_n)\in B_n\setminus(B_n\cap A(n,\delta))} P_n(x_1,\ldots,x_n)2^{-n(D(P||Q)+\delta)} \\ &\geq (1-2\epsilon)2^{-n(D(P||Q)-\delta)} + (1-(1-2\epsilon)-\epsilon)2^{-n(D(P||Q)+\delta)} \text{ (This step seems wrong}\end{aligned}$$

Let's follow the image again.

$$s_q(n,\epsilon) = \sum_{(x_1,\ldots,x_n)\in B_n\cap A(n,\delta)} Q_n(\cdot) + \sum_{(x_1,\ldots,x_n)\in B_n\setminus(B_n\cap A(n,\delta))} Q_n(\cdot)$$

We know $Q_n(\cdot) \geq P_n(\cdot)2^{-n(D(P||Q)+\delta)}$ for the second term.

$$s_q(n,\epsilon) \geq (1-2\epsilon)2^{-n(D(P||Q)-\delta)} + \epsilon \cdot 2^{-n(D(P||Q)+\delta)} \text{ (Using probability bounds)}$$

This still doesn't match the image. Let's look at the end of the image.

$$2^{n(D(P||Q)+\epsilon)}\left[\sum_{(x_1,\ldots,x_n)\in B_n\cap A(n,\delta)} Q_n(x_1,\ldots,x_n) + \sum_{(x_1,\ldots,x_n)\in A(n,\delta)^c\cap B_n} Q_n(x_1,\ldots,x_n)\right] \geq 1-\epsilon$$

The term in the bracket is $s_q(n,\epsilon)$.

$$s_q(n,\epsilon) \geq (1-\epsilon)2^{-n(D(P||Q)+\epsilon)}$$

$$\frac{\log s_q(n,\epsilon)}{n} \geq \frac{\log(1-\epsilon)}{n} - (D(P||Q)+\epsilon)$$

$$\liminf_{n\to\infty} \frac{\log s_q(n,\epsilon)}{n} \geq -(D(P||Q)+\epsilon)$$

As $\epsilon \to 0$, we get $\liminf_{n\to\infty} \frac{\log s_q(n,\epsilon)}{n} \geq -D(P||Q)$.

The diagram shows $P(B_n) \geq 1-\epsilon$ and $P(A_n) \geq 1-\epsilon$. The intersection has probability at least $1-2\epsilon$. The region in B_n but outside A_n has probability at most ϵ.

1) Achievability:

$$A(n,\delta) = \left\{(x_1,\ldots,x_n) \in \mathcal{A}^n : \left|\frac{1}{n}\log\frac{P_n(x_1,\ldots,x_n)}{Q_n(x_1,\ldots,x_n)} - D(P||Q)\right| < \epsilon\right\}$$

Let the sequences in $A(n,\delta)$ be encoded one-to-one to a unique codeword.

$$s_q(n,\epsilon) = \min_{T:P(T)\text{ P-probability }\geq 1-\epsilon} Q\text{-wt of sets}$$

$$s_q(n,\epsilon) \leq Q(A(n,\delta)) \quad \cdots(1)$$

as $P(A(n,\delta)) \geq 1-\epsilon$.

We know that,

$$\left|\frac{1}{n}\log\frac{P_n(x_1,\ldots,x_n)}{Q_n(x_1,\ldots,x_n)} - D(P||Q)\right| < \epsilon$$

$$-\epsilon < \frac{1}{n}\log\frac{P_n(x_1,\ldots,x_n)}{Q_n(x_1,\ldots,x_n)} - D(P||Q) < \epsilon$$

$$D(P||Q) - \epsilon < \frac{1}{n}\log\frac{P_n(x_1,\ldots,x_n)}{Q_n(x_1,\ldots,x_n)} < D(P||Q) + \epsilon$$

$$
\begin{aligned}
\log \frac{P_n(x_1,\ldots,x_n)}{Q_n(x_1,\ldots,x_n)} &< nD(P||Q) + n\epsilon \\
\frac{P_n(x_1,\ldots,x_n)}{Q_n(x_1,\ldots,x_n)} &< 2^{nD(P||Q)+n\epsilon}
\end{aligned}
$$

Similarly, the other way:

$$2^{n(D(P||Q))-n\delta} < \frac{P_n(x_1,\ldots,x_n)}{Q_n(x_1,\ldots,x_n)} < 2^{n(D(P||Q))+n\delta}$$

This implies:

$$
\begin{aligned}
Q_n(x_1,\ldots,x_n) &< P_n(x_1,\ldots,x_n)2^{-nD(P||Q)+n\delta} \\
Q_n(x_1,\ldots,x_n) &> P_n(x_1,\ldots,x_n)2^{-nD(P||Q)-n\delta}
\end{aligned}
$$

From the previous steps, we know that for $x^n \in A(n,\delta)$, $Q_n(x_1,\ldots,x_n) < P_n(x_1,\ldots,x_n)2^{-nD(P||Q)+n\delta}$. Now, let's find an upper bound for $Q(A(n,\delta))$:

$$
\begin{aligned}
Q(A(n,\delta)) &= \sum_{(x_1,\ldots,x_n)\in A(n,\delta)} Q_n(x_1,\ldots,x_n) \\
&\leq \sum_{(x_1,\ldots,x_n)\in A(n,\delta)} P_n(x_1,\ldots,x_n)2^{-nD(P||Q)+n\delta} \\
&= 2^{-nD(P||Q)+n\delta} \sum_{(x_1,\ldots,x_n)\in A(n,\delta)} P_n(x_1,\ldots,x_n) \\
&\leq 2^{-nD(P||Q)+n\delta} \cdot 1
\end{aligned}
$$

So,

$$Q(A(n,\delta)) \leq 2^{-nD(P||Q)+n\delta} \quad \cdots (2)$$

From (1) and (2), we have:

$$s_q(n,\epsilon) \leq Q(A(n,\delta)) \leq 2^{-nD(P||Q)+n\delta}$$

$$\frac{\log s_q(n,\epsilon)}{n} \leq -D(P||Q) + \delta$$

Taking the limit superior as $n \to \infty$:

$$\limsup_{n\to\infty} \frac{\log s_q(n,\epsilon)}{n} \leq -D(P||Q) + \delta$$

Since this holds for any $\delta > 0$, we can take $\delta \to 0$ to get:

$$\limsup_{n\to\infty} \frac{\log s_q(n,\epsilon)}{n} \leq -D(P||Q)$$

We have, from the previous steps:

$$\liminf_{n\to\infty} \frac{\log s_q(n,\epsilon)}{n} \geq -D(P||Q) - \delta$$

And,

$$\limsup_{n\to\infty} \frac{\log s_q(n,\epsilon)}{n} \leq -D(P||Q) + \delta$$

We also know that for any sequence, the limit inferior is always less than or equal to the limit superior:

$$\liminf_{n\to\infty} \frac{\log s_q(n,\epsilon)}{n} \leq \limsup_{n\to\infty} \frac{\log s_q(n,\epsilon)}{n}$$

Combining all these inequalities, we get:

$$-D(P||Q) - \delta \leq \liminf_{n\to\infty} \frac{\log s_q(n,\epsilon)}{n} \leq \limsup_{n\to\infty} \frac{\log s_q(n,\epsilon)}{n} \leq -D(P||Q) + \delta$$

As δ is arbitrary, the limit inferior and the limit superior must converge to the same value:

$$\lim_{n\to\infty} \frac{\log s_q(n,\epsilon)}{n} = -D(P||Q)$$

(Assignment-2.2#324)

Suppose a_i and b_i are positive numbers for $i = 1, \ldots, n$. Prove the log-sum inequality:

$$\sum_i a_i \log \frac{a_i}{b_i} \geq \left(\sum_i a_i\right) \log \frac{\sum_i a_i}{\sum_i b_i}.$$

Let $x_i = \frac{a_i}{b_i}$ for $i = 1, \ldots, n$. Let $B = \sum_{j=1}^n b_j$. Define weights $w_i = \frac{b_i}{B}$ for $i = 1, \ldots, n$. Since $b_i > 0$ and $B > 0$, we have $w_i > 0$. Also, $\sum_{i=1}^n w_i = \sum_{i=1}^n \frac{b_i}{B} = \frac{1}{B}\sum_{i=1}^n b_i = \frac{B}{B} = 1$. Thus, w_i form a probability distribution over the index i.

Now, let's rewrite the left-hand side (LHS) of the inequality:

$$\text{LHS} = \sum_{i=1}^n a_i \log \frac{a_i}{b_i} = \sum_{i=1}^n (x_i b_i) \log x_i$$

Since $b_i = Bw_i$, we have:

$$\text{LHS} = \sum_{i=1}^n x_i(Bw_i) \log x_i = B \sum_{i=1}^n w_i(x_i \log x_i)$$

Let X be a random variable taking values x_i with probabilities w_i. Then LHS $= BE[X \log X]$.

Now, let's look at the right-hand side (RHS). Let $A = \sum_{i=1}^n a_i$. We can write A in terms of x_i and w_i:

$$A = \sum_{i=1}^n a_i = \sum_{i=1}^n x_i b_i = \sum_{i=1}^n x_i(Bw_i) = B\sum_{i=1}^n w_i x_i = BE[X].$$

The term inside the logarithm on the RHS is $\frac{A}{B} = \frac{BE[X]}{B} = E[X]$. So, the RHS is:

$$\text{RHS} = A \log \frac{A}{B} = (BE[X]) \log(E[X])$$

The inequality we want to prove is now $BE[X \log X] \geq BE[X] \log(E[X])$. Since $B = \sum b_i > 0$, this is equivalent to:

$$E[X \log X] \geq E[X] \log(E[X])$$

We will use Jensen's inequality, which states that for a convex function $g(x)$ and a random variable X with expectation $E[X]$, we have $E[g(X)] \geq g(E[X])$.

Let $g(x) = x \log_b x = x \frac{\ln x}{\ln b}$ for $x > 0$. We find the second derivative:

$$g'(x) = \frac{\ln x + 1}{\ln b}$$

$$g''(x) = \frac{1}{x \ln b}$$

Since $x > 0$ and $b > 1$ (so $\ln b > 0$), we have $g''(x) > 0$ for $x > 0$. Therefore, $g(x) = x \log x$ is a convex function for $x > 0$.

Applying Jensen's inequality to $g(X) = X \log X$ with expectation E with respect to the weights w_i, we get:

$$E[X \log X] \geq (E[X]) \log(E[X])$$

This proves the log-sum inequality.

(Assignment-2.3#325)

Find the derivative of $f(p) = -p \log p - (1-p) \log(1-p)$ at $p = 0$ and at $p = 1$.

We are asked to find the derivative of the function $f(p) = -p \log p - (1-p) \log(1-p)$ at $p = 0$ and at $p = 1$. We assume that log refers to the logarithm with base $b > 1$.

First, we find the derivative of $f(p)$ with respect to p for $p \in (0, 1)$:

$$f'(p) = \frac{d}{dp}(-p \log p) - \frac{d}{dp}((1-p) \log(1-p))$$

Using the product rule and the chain rule, we found that:

$$f'(p) = \log(1-p) - \log p = \log\left(\frac{1-p}{p}\right)$$

Now, we need to consider the limits of the derivative as p approaches 0 and 1 from within the interval $(0, 1)$.

At $p = 0$:

$$f'(0) = \lim_{p \to 0^+} f'(p) = \lim_{p \to 0^+} \log_b \left(\frac{1-p}{p} \right)$$

As $p \to 0^+$, $1 - p \to 1$, so $\frac{1-p}{p} \to \frac{1}{0^+} = +\infty$. Since $b > 1$, $\lim_{x \to +\infty} \log_b x = +\infty$. Therefore, the derivative at $p = 0$ is $+\infty$.

At $p = 1$:

$$f'(1) = \lim_{p \to 1^-} f'(p) = \lim_{p \to 1^-} \log_b \left(\frac{1-p}{p} \right)$$

As $p \to 1^-$, $1 - p \to 0^+$, and $p \to 1$, so $\frac{1-p}{p} \to \frac{0^+}{1} = 0^+$. Since $b > 1$, $\lim_{x \to 0^+} \log_b x = -\infty$. Therefore, the derivative at $p = 1$ is $-\infty$.

Conclusion: The derivative of $f(p) = -p \log p - (1 - p) \log(1 - p)$ at $p = 0$ is $+\infty$, and at $p = 1$ is $-\infty$ (in the sense of limits).

(Assignment-2.4#326)

Show that instantaneous codes are uniquely decodable.

Proof: We will prove this by contradiction. Assume that an instantaneous code $C = \{c_1, c_2, \ldots, c_n\}$ for a source alphabet $S = \{s_1, s_2, \ldots, s_n\}$ is not uniquely decodable. This means there exists at least one encoded sequence y that can be decoded into two different sequences of source symbols. Let these two sequences of source symbols be $s_{i_1}, s_{i_2}, \ldots, s_{i_k}$ and $s_{j_1}, s_{j_2}, \ldots, s_{j_m}$, where the two sequences of indices $(i_1, \ldots, i_k)$ and $(j_1, \ldots, j_m)$ are not identical. The corresponding encoded sequence is:

$$y = c_{i_1} c_{i_2} \ldots c_{i_k} = c_{j_1} c_{j_2} \ldots c_{j_m}$$

First, let's argue that the lengths of the two decoded sequences must be the same, i.e., $k = m$. Assume, without loss of generality, that $k < m$. Let l be the first index where $i_l \neq j_l$. If no such index exists up to k, it means $i_r = j_r$ for $r = 1, \ldots, k$, so $c_{i_r} = c_{j_r}$ for $r = 1, \ldots, k$. Then we have $c_{i_1} \ldots c_{i_k} = c_{j_1} \ldots c_{j_k} = c_{j_1} \ldots c_{j_m}$. This implies that the remaining part $c_{j_{k+1}} \ldots c_{j_m}$ must be an empty string. However, since codewords in a code are usually assumed to be non-empty, this leads to a contradiction as $m > k$ means there is at least one non-empty codeword in $c_{j_{k+1}} \ldots c_{j_m}$. Thus, we must have $k = m$.

Now, since the two decoded sequences are different and have the same length $k = m$, there must be a first index l $(1 \leq l \leq k)$ where they differ, i.e., $i_l \neq j_l$. This implies that the codewords at this position are also different, $c_{i_l} \neq c_{j_l}$. However, the codewords before this position must be the same: $c_{i_r} = c_{j_r}$ for $1 \leq r < l$ (if $l > 1$). Now, consider the remaining parts of the encoded sequence starting from index l:

$$c_{i_l} c_{i_{l+1}} \dots c_{i_k} = c_{j_l} c_{j_{l+1}} \dots c_{j_k}$$

Since c_{i_l} and c_{j_l} are different codewords that start at the same position in the remaining sequence, one of them must be a prefix of the other. If c_{i_l} is a prefix of c_{j_l}, this contradicts the definition of an instantaneous code. If c_{j_l} is a prefix of c_{i_l}, this also contradicts the definition of an instantaneous code. Therefore, our initial assumption that the instantaneous code is not uniquely decodable must be false.

Conclusion: Instantaneous codes are uniquely decodable.

(Assignment-2.5#327)

If $g : \mathcal{A} \to \mathcal{B}$, show that $H(X) = H(g(X))$ iff g is invertible.

We need to prove the statement in both directions:

(i) ($\implies$) Assume $H(X) = H(g(X))$. We need to show that g is invertible.

Let $Y = g(X)$. The entropy of X is given by

$$H(X) = -\sum_{x \in \mathcal{A}} P(X = x) \log P(X = x)$$

and the entropy of $Y = g(X)$ is given by

$$H(Y) = H(g(X)) = -\sum_{y \in \mathcal{B}} P(Y = y) \log P(Y = y)$$

We are given that $H(X) = H(g(X))$.

Consider the conditional entropy $H(X|Y) = H(X|g(X))$. By the chain rule for entropy, we have:

$$H(X, Y) = H(X) + H(Y|X) = H(Y) + H(X|Y)$$

Since $Y = g(X)$, for a given value of $X = x$, the value of $Y = g(x)$ is uniquely determined. Therefore, $P(Y = y|X = x) = 1$ if $y = g(x)$ and 0 otherwise. This implies $H(Y|X) = 0$.

Substituting $H(Y|X) = 0$ into the chain rule, we get:

$$H(X, Y) = H(X)$$

Now, using the other part of the chain rule and the given condition $H(X) = H(Y)$, we have:

$$H(X) = H(Y) + H(X|Y) = H(X) + H(X|Y)$$

This implies that $H(X|Y) = 0$.

The conditional entropy $H(X|Y) = 0$ if and only if X is completely determined by Y. That is, for every value $y \in \mathcal{B}$ that Y can take with non-zero probability, there is a unique value $x \in \mathcal{A}$ such that $g(x) = y$.

If for some $y \in \mathcal{B}$ with $P(Y = y) > 0$, there were two distinct values $x_1, x_2 \in \mathcal{A}$ such that $g(x_1) = y$ and $g(x_2) = y$, then when $Y = y$ is observed, we would not be able to uniquely determine whether $X = x_1$ or $X = x_2$, leading to $H(X|Y) > 0$, which contradicts our finding.

Therefore, for every y in the range of g, there is a unique $x \in \mathcal{A}$ such that $g(x) = y$. This means that g is injective (one-to-one) on the set of values that X can take with non-zero probability.

Now, let's consider the sizes of the alphabets. Since $Y = g(X)$, the support of Y (the set of values y with $P(Y = y) > 0$) is a subset of $\mathcal{B}$, and there is a one-to-one correspondence between the support of X and the support of Y. For the entropies to be equal, the number of elements with non-zero probability in $\mathcal{A}$ and $\mathcal{B}$ (specifically, the support of X and Y) must be the same.

If g is not surjective (onto), then $|\text{Range}(g)| < |\mathcal{B}|$. Since there is a one-to-one mapping from the support of X to the support of $Y = g(X)$, the size of the support of X must be equal to the size of the support of Y. If $|\mathcal{A}| > |\mathcal{B}|$, then g cannot be injective. If $|\mathcal{A}| < |\mathcal{B}|$, it is possible for g to be injective but not surjective. However, if $H(X) = H(g(X))$, the number of outcomes with non-zero probability must be the same. If g is not surjective, there would be some $b \in \mathcal{B}$ such that $P(Y = b) = 0$. This doesn't directly contradict $H(X) = H(Y)$, but the injectivity derived from $H(X|Y) = 0$ implies that the number of elements in the support of X is equal to the number of elements in the support of Y. For g to be invertible, it must be a bijection from $\mathcal{A}$ to $\mathcal{B}$.

If g is not a bijection, then either it's not injective or not surjective. We showed that $H(X) = H(g(X))$ implies injectivity on the support of X. If $|\mathcal{A}| > |\mathcal{B}|$, g cannot be injective. If $|\mathcal{A}| < |\mathcal{B}|$, and g is injective, then the size of the support of Y would be equal to the size of the support of X, but g would not be surjective. However, if $H(X) = H(g(X))$, the spread of probabilities must be preserved, implying a one-to-one correspondence between the elements with non-zero probability. If $|\mathcal{A}| \neq |\mathcal{B}|$, it's hard to maintain this equality for all distributions of X.

Let's refine the argument. $H(X|Y) = 0$ implies that X is a function of Y, say $X = h(Y)$. Since $Y = g(X)$, we have $X = h(g(X))$ for all x in the support of X. Also, $Y = g(h(Y))$ for all y in the support of Y. This means that g has a left inverse h on the support of Y and h has a left inverse g on the support of X. This implies that g is a bijection between the support of X and the support of Y. If the support of X is $\mathcal{A}$ and the support of Y is $\mathcal{B}$, then g is a bijection from $\mathcal{A}$ to $\mathcal{B}$, and hence invertible. For this to hold for any distribution of X, we must have $|\mathcal{A}| = |\mathcal{B}|$.

(ii) ($\Longleftarrow$) Assume g is invertible. We need to show that $H(X) = H(g(X))$.

Since g is invertible, it is a bijection from $\mathcal{A}$ to $\mathcal{B}$. Let $y = g(x)$. Then $x = g^{-1}(y)$. The probability mass function of $Y = g(X)$ is given by:

$$P(Y = y) = P(g(X) = y) = P(X = g^{-1}(y))$$

Let $x = g^{-1}(y)$. Then $P(Y = y) = P(X = x)$.

Now, let's compute the entropy of Y:

$$H(Y) = -\sum_{y \in \mathcal{B}} P(Y = y) \log P(Y = y)$$

Since g is a bijection, for every $y \in \mathcal{B}$, there is a unique $x \in \mathcal{A}$ such that $y = g(x)$, and $P(Y = g(x)) = P(X = x)$. We can rewrite the sum over $y \in \mathcal{B}$ as a sum over the corresponding $x \in \mathcal{A}$:

$$H(Y) = -\sum_{x \in \mathcal{A}} P(g(X) = g(x)) \log P(g(X) = g(x))$$

$$H(Y) = -\sum_{x \in \mathcal{A}} P(X = x) \log P(X = x)$$

This is exactly the definition of the entropy of X:

$$H(Y) = H(X)$$

Thus, if g is invertible, then $H(X) = H(g(X))$.

Conclusion:

We have shown that $H(X) = H(g(X))$ if and only if g is invertible.

(Assignment-2.6#328)

Prove or disprove: "$H(X|Y = y)$ can be strictly larger than $H(X)$."

$H(X|Y) \leq H(X)$. But for specific values of Y, we can have $H(X|Y = b) > H(X)$, which means knowledge of Y has resulted in the increase in uncertainty of X. This counterintuitive phenomenon occurs when the conditional distribution $P(XY = b)$ is more dispersed or uniform than the original distribution $P(X)$, leading to greater unpredictability in X. Example:

Given Joint Probabilities:

- $P(X = 0, Y = 1) = 0.5$
- $P(X = 1, Y = 1) = 0.25$

- $P(X = 0, Y = 2) = 0.25$
- $P(X = 1, Y = 2) = 0$

First, we need to find the marginal probabilities of X:

- $P(X = 0) = P(X = 0, Y = 1) + P(X = 0, Y = 2) = 0.5 + 0.25 = 0.75$
- $P(X = 1) = P(X = 1, Y = 1) + P(X = 1, Y = 2) = 0.25 + 0 = 0.25$

Now we can calculate the entropy of X:

$$\begin{aligned}
H(X) &= -\sum_{x\in\{0,1\}} P(X = x)\log_2 P(X = x) \\
&= -\left[P(X = 0)\log_2 P(X = 0) + P(X = 1)\log_2 P(X = 1)\right] \\
&= -\left[0.75\log_2 0.75 + 0.25\log_2 0.25\right] \\
&\approx -\left[0.75\cdot(-0.415) + 0.25\cdot(-2)\right] \\
&\approx -\left[-0.311 - 0.5\right] \\
&\approx 0.811
\end{aligned}$$

Calculation of H(X|Y=1):

To calculate $H(X|Y = 1)$, we first need the conditional probabilities $P(X = x|Y = 1)$:

- $P(X = 0|Y = 1) = \frac{P(X=0,Y=1)}{P(Y=1)}$
- $P(X = 1|Y = 1) = \frac{P(X=1,Y=1)}{P(Y=1)}$

We need to calculate P(Y=1): $P(Y = 1) = P(X = 0, Y = 1) + P(X = 1, Y = 1) = 0.5 + 0.25 = 0.75$

Now we can calculate the conditional probabilities:

- $P(X = 0|Y = 1) = \frac{0.5}{0.75} = \frac{2}{3}$
- $P(X = 1|Y = 1) = \frac{0.25}{0.75} = \frac{1}{3}$

Finally, we calculate $H(X|Y=1)$:

$$\begin{aligned}
H(X|Y=1) &= -\sum_{x\in\{0,1\}} P(X=x|Y=1)\log_2 P(X=x|Y=1) \\
&= -\left[P(X=0|Y=1)\log_2 P(X=0|Y=1) + P(X=1|Y=1)\log_2 P(X=1|Y=1)\right] \\
&= -\left[\frac{2}{3}\log_2\frac{2}{3} + \frac{1}{3}\log_2\frac{1}{3}\right] \\
&\approx -\left[\frac{2}{3}\cdot(-0.585) + \frac{1}{3}\cdot(-1.585)\right] \\
&\approx -\left[-0.390 - 0.528\right] \\
&\approx 0.918
\end{aligned}$$

Results:

- $H(X) \approx 0.811$
- $H(X|Y=1) \approx 0.918$

Clearly, we see that locally, for a specific $Y=1$, the conditional entropy turns out to be larger.

(Assignment-2.7#329)

Entropy of functions of a random variable. Let X be a discrete random variable. Show that the entropy of a function of X is less than or equal to the entropy of X by justifying the following steps:

$$\begin{aligned}
H(X,g(X)) &= H(X) + H(g(X)|X) \\
&= H(X), \\
H(X,g(X)) &= H(g(X)) + H(X|g(X)) \\
&\geq H(g(X)).
\end{aligned}$$

Thus, $H(g(X)) \leq H(X)$.

We are given a discrete random variable X and a function of X, denoted as $g(X)$. We want to show that the entropy of the function, $H(g(X))$, is less than or equal to the entropy of the random variable itself, $H(X)$. We will justify the given steps:

Step (a): $H(X,g(X)) = H(X) + H(g(X)|X)$

The chain rule for entropy states that for any two random variables A and B, $H(A,B) = H(A) + H(B|A)$. Substituting X for A and $g(X)$ for B gives us the equation in step (a).

Step (b): $H(X) + H(g(X)|X) = H(X)$

Since $g(X)$ is a function of X, the value of $g(X)$ is completely determined by the value of X. Therefore, knowing X removes all uncertainty about $g(X)$. By the definition of conditional entropy, if Y is a function of X, then $H(Y|X) = 0$. Hence, $H(g(X)|X) = 0$, and $H(X) + H(g(X)|X) = H(X) + 0 = H(X)$.

Step (c): $H(X, g(X)) = H(g(X)) + H(X|g(X))$

This is again the chain rule for entropy, but with the order of X and $g(X)$ reversed.

Step (d): $H(g(X)) + H(X|g(X)) \geq H(g(X))$

Entropy is always non-negative. Conditional entropy, by definition, is also non-negative. Therefore, $H(X|g(X)) \geq 0$. Adding a non-negative quantity to $H(g(X))$ will result in a value greater than or equal to $H(g(X))$. Hence, $H(g(X)) + H(X|g(X)) \geq H(g(X))$.

Conclusion:

From steps (a) and (b), we have $H(X, g(X)) = H(X)$. From steps (c) and (d), we have $H(X, g(X)) \geq H(g(X))$.

Combining these results, we get $H(X) = H(X, g(X)) \geq H(g(X))$.

Therefore, $H(g(X)) \leq H(X)$. This shows that the entropy of a function of a random variable is less than or equal to the entropy of the random variable itself.

(Assignment-2.8#330)

Zero conditional entropy. Show that if $H(Y|X) = 0$, then Y is a function of X [i.e., for all x with $p(x) > 0$, there is only one possible value of y with $p(x, y) > 0$].

Recall the definition of conditional entropy:

$$H(Y|X) = \sum_x p(x) H(Y|X = x)$$

where

$$H(Y|X = x) = -\sum_y p(y|x) \log_2 p(y|x)$$

Since $H(Y|X) = 0$, we have:

$$0 = \sum_x p(x) H(Y|X = x)$$

We know that $p(x) \geq 0$ for all x, and $H(Y|X = x) \geq 0$ for all x (because entropy is always non-negative). Therefore, the sum of non-negative terms can only be zero if each term in the sum is zero.

Thus, for all x with $p(x) > 0$, we must have:

$$H(Y|X = x) = 0$$

Now, let's consider what $H(Y|X = x) = 0$ implies.

$$0 = -\sum_{y} p(y|x) \log_2 p(y|x)$$

Again, we have a sum of non-negative terms (since $p(y|x) \geq 0$ and $-\log_2 p(y|x) \geq 0$) that equals zero. This can only happen if for all y, either $p(y|x) = 0$ or $\log_2 p(y|x) = 0$ (which implies $p(y|x) = 1$).

Therefore, for a given x, we have the following possibilities for each y:

- $p(y|x) = 0$
- $p(y|x) = 1$

Since $\sum_y p(y|x) = 1$, if $p(y|x) = 1$ for some y, then $p(y'|x) = 0$ for all $y' \neq y$. This means that given x, there is only one y with $p(y|x) > 0$, and for that particular y, $p(y|x) = 1$.

In other words, for each x with $p(x) > 0$, there is only one possible value of y such that $p(x, y) > 0$. This is precisely the definition of Y being a function of X.

Conclusion: If $H(Y|X) = 0$, then Y is a function of X.

(Assignment-2.9#331)

Entropy of a sum. Let X and Y be random variables that take on values $x_1, x_2, \ldots, x_r$ and $y_1, y_2, \ldots, y_s$, respectively. Let $Z = X + Y$.

(a) Show that $H(Z|X) = H(Y|X)$. Argue that if X, Y are independent, then $H(Y) \leq H(Z)$ and $H(X) \leq H(Z)$. Thus, the addition of independent random variables adds uncertainty.

(b) Give an example of (necessarily dependent) random variables in which $H(X) > H(Z)$ and $H(Y) > H(Z)$.

(c) Under what conditions does $H(Z) = H(X) + H(Y)$?

(a) We want to show that $H(Z|X) = H(Y|X)$ where $Z = X + Y$. The conditional entropy $H(Z|X)$ is defined as:

$$H(Z|X) = -\sum_x \sum_z p(x,z) \log_2 p(z|x)$$

Since $Z = X+Y$, for a given value of $X = x$, the value of Z is $z = x+y$, which corresponds to a unique value of $Y = y = z - x$. Thus, the mapping between y and z (given x) is one-to-one. We can change the summation over z to a summation over y:

$$H(Z|X) = -\sum_x \sum_y p(x, x+y) \log_2 p(x+y|x)$$

We know that $p(x+y|x) = \frac{p(x,x+y)}{p(x)}$. Also, the joint probability $p(x, x+y)$ is the same as $p(x,y)$ because for a given x, y uniquely determines $x+y$ and vice versa. Therefore, $p(x+y|x) = \frac{p(x,y)}{p(x)} = p(y|x)$. Substituting this back, we get:

$$H(Z|X) = -\sum_x \sum_y p(x,y) \log_2 p(y|x)$$

$$H(Z|X) = -\sum_{x,y} p(x,y) \log_2 p(y|x) = H(Y|X)$$

Thus, $H(Z|X) = H(Y|X)$.

Now, if X and Y are independent, then $H(Y|X) = H(Y)$. From the above result, $H(Z|X) = H(Y)$. We know that conditioning reduces entropy, i.e., $H(Z) \geq H(Z|X)$. Therefore, $H(Z) \geq H(Y)$. By symmetry, let's consider $H(Z|Y) = H(X|Y)$ (since $X = Z - Y$). If X and Y are independent, then $H(X|Y) = H(X)$, so $H(Z|Y) = H(X)$. And again, $H(Z) \geq H(Z|Y)$, which implies $H(Z) \geq H(X)$. Thus, if X and Y are independent, then $H(Y) \leq H(Z)$ and $H(X) \leq H(Z)$, suggesting that the addition of independent random variables adds uncertainty (in terms of entropy).

(b) Let X be a random variable taking values in $\{0,1\}$ with $P(X=0) = 0.5$ and $P(X=1) = 0.5$. Then $H(X) = 1$ bit. Let $Y = -X$. Y takes values in $\{0,-1\}$ with $P(Y=0) = 0.5$ and $P(Y=-1) = 0.5$, so $H(Y) = 1$ bit. $Z = X+Y = X-X = 0$. Z is a constant with value 0, so $H(Z) = 0$ bits. In this example, $H(X) = 1 > 0 = H(Z)$ and $H(Y) = 1 > 0 = H(Z)$. Note that X and Y are dependent here (Y is completely determined by X).

(c) We want to find the conditions under which $H(Z) = H(X) + H(Y)$ where $Z = X + Y$. We know that for any two random variables X and Y, the joint entropy satisfies $H(X,Y) \leq H(X) + H(Y)$, with equality if and only if X and Y are independent. We also know that $H(Z) \leq H(X,Z)$. Since $Z = X + Y$, the pair $(X,Z) = (X, X+Y)$ has a one-to-one correspondence with (X,Y) (as $Y = Z - X$). Therefore, $H(X,Z) = H(X,Y)$. Combining these, we have $H(Z) \leq H(X,Y) \leq H(X) + H(Y)$. For $H(Z) = H(X) + H(Y)$ to hold, we must have equality throughout. This requires two conditions:

1. $H(X,Y) = H(X) + H(Y)$, which implies that X and Y are independent.

2. $H(Z) = H(X,Y)$, which implies that the sum $Z = X+Y$ retains all the information of the pair (X,Y). This occurs if and only if the mapping from (X,Y) to Z is one-to-one (on the support where the probability mass function is non-zero).

The second condition is met if and only if at least one of X or Y is a constant (has zero entropy). For example, if $Y = c$ with probability 1, then $Z = X + c$, and $H(Z) = H(X)$. In this case, $H(X) + H(Y) = H(X) + 0 = H(X) = H(Z)$.

Thus, the condition under which $H(Z) = H(X) + H(Y)$ is that X and Y are independent, and at least one of them has zero entropy (i.e., is a constant).

(Assignment-2.10#332)

Infinite entropy. This problem shows that the entropy of a discrete random variable can be infinite. Let $A = \sum_{n=2}^{\infty}(n \log^2 n)^{-1}$. [It is easy to show that A is finite by bounding the infinite sum by the integral of $(x \log^2 x)^{-1}$.] Show that the integer-valued random variable X defined by $\Pr(X = n) = (An \log^2 n)^{-1}$ for $n = 2, 3, \ldots$, has $H(X) = +\infty$.

The entropy of a discrete random variable X with probability mass function $p_n = \Pr(X = n)$ is defined as

$$H(X) = -\sum_n p_n \log_2 p_n$$

In this problem, the random variable X takes values in the set $\{2, 3, \ldots\}$ with probabilities $p_n = \frac{1}{An \log^2 n}$. Substituting this into the entropy formula, we get:

$$H(X) = -\sum_{n=2}^{\infty} \left(\frac{1}{An \log^2 n}\right) \log_2 \left(\frac{1}{An \log^2 n}\right)$$

$$H(X) = -\sum_{n=2}^{\infty} \frac{1}{An \log^2 n} \left(-\log_2(An \log^2 n)\right)$$

$$H(X) = \frac{1}{A} \sum_{n=2}^{\infty} \frac{\log_2(An \log^2 n)}{n \log^2 n}$$

We can rewrite the logarithm term using the properties of logarithms:

$$\log_2(An \log^2 n) = \log_2 A + \log_2 n + \log_2(\log^2 n) = \log_2 A + \log_2 n + 2\log_2(\log n)$$

Substituting this back into the expression for $H(X)$:

$$H(X) = \frac{1}{A} \sum_{n=2}^{\infty} \frac{\log_2 A + \log_2 n + 2\log_2(\log n)}{n \log^2 n}$$

We can split this sum into three parts:

$$H(X) = \frac{1}{A}\left(\sum_{n=2}^{\infty} \frac{\log_2 A}{n\log^2 n} + \sum_{n=2}^{\infty} \frac{\log_2 n}{n\log^2 n} + \sum_{n=2}^{\infty} \frac{2\log_2(\log n)}{n\log^2 n}\right)$$

$$H(X) = \frac{1}{A}\left(\log_2 A \sum_{n=2}^{\infty} \frac{1}{n\log^2 n} + \sum_{n=2}^{\infty} \frac{1}{n\log n} + 2\sum_{n=2}^{\infty} \frac{\log_2(\log n)}{n\log^2 n}\right)$$

We know that $A = \sum_{n=2}^{\infty}(n\log^2 n)^{-1}$ is a finite positive constant. Therefore, the first term $\frac{\log_2 A}{A}\sum_{n=2}^{\infty}\frac{1}{n\log^2 n} = \log_2 A$ is a finite value.

Consider the second term: $\sum_{n=2}^{\infty}\frac{1}{n\log n}$. We can use the integral test to determine its convergence. Consider the integral

$$\int_2^{\infty} \frac{1}{x\log x} dx$$

Let $u = \log x$, then $du = \frac{1}{x}dx$. When $x = 2$, $u = \log 2$. As $x \to \infty$, $u \to \infty$. The integral becomes:

$$\int_{\log 2}^{\infty} \frac{1}{u} du = [\ln u]_{\log 2}^{\infty} = \lim_{b\to\infty}(\ln b - \ln(\log 2)) = +\infty$$

Since the integral diverges, the sum $\sum_{n=2}^{\infty}\frac{1}{n\log n}$ also diverges to $+\infty$.

Now consider the third term: $2\sum_{n=2}^{\infty}\frac{\log_2(\log n)}{n\log^2 n}$. For sufficiently large n, $\log n > 1$, so $\log_2(\log n) > 0$. However, the growth of $\log_2(\log n)$ is much slower than $\log n$. We can compare this sum with $\sum \frac{1}{n\log^{1+\epsilon} n}$ which converges for any $\epsilon > 0$. In this case, the $\log^2 n$ in the denominator will dominate the $\log_2(\log n)$ in the numerator, suggesting this sum might converge. However, we already have a term that diverges to infinity.

Since $H(X)$ contains the term $\frac{1}{A}\sum_{n=2}^{\infty}\frac{1}{n\log n}$, which diverges to $+\infty$, and A is a finite positive constant, the entropy $H(X)$ is indeed infinite.

$$H(X) = \frac{\log_2 A}{A}\sum_{n=2}^{\infty}\frac{1}{n\log^2 n} + \frac{1}{A}\sum_{n=2}^{\infty}\frac{1}{n\log n} + \frac{2}{A}\sum_{n=2}^{\infty}\frac{\log_2(\log n)}{n\log^2 n} = +\infty$$

(Assignment-3.1#333)

1. Convergence of Cesaro mean. Let $\lim_{n\to\infty} a_n = a$. Let $b_n = \frac{1}{n}\sum_{i=1}^{n} a_i$. Then show that $\lim_{n\to\infty} b_n = a$. Identify a counterexample where $\lim_{n\to\infty} b_n$ exists but $\lim_{n\to\infty} a_n$ does not.

(Assignment-3.2#334)

Let $\{s_n\}$ be a nonnegative sequence such that $0 \le s_n < \infty$ for every n. Suppose that the sequence is subadditive, i.e., for any two natural numbers k, l, we have $s_{k+l} \le s_k + s_l$. Then show that $\lim_{n\to\infty} \frac{s_n}{n}$ exists and equals $\inf_{n\ge1} \frac{s_n}{n}$. [Hint: As a first step, group into blocks of size m and use subadditivity to show $\limsup_{n\to\infty} \frac{s_n}{n} \le \frac{s_m}{m}$.]

(Assignment-3.3#335)

3. Time's arrow. Let $(X_i)_{i=-\infty}^{\infty}$ be a stationary stochastic process. Prove that

$$H(X_0|X_{-1}, X_{-2}, \ldots, X_{-n}) = H(X_0|X_1, X_2, \ldots, X_n).$$

In other words, the present has a conditional entropy given the past equal to the conditional entropy given the future. This is true even though it is quite easy to concoct stationary random processes for which the flow into the future looks quite different from the flow into the past. That is, one can determine the direction of time by looking at a sample function of the process. Nevertheless, given the present state, the conditional uncertainty of the next symbol in the future is equal to the conditional uncertainty of the previous symbol in the past.

(Assignment-3.4#336)

Pairwise independence. Let $X_1, X_2, \ldots, X_{n-1}$ be i.i.d. random variables taking values in $\{0, 1\}$ with $\Pr\{X_i = 1\} = \frac{1}{2}$. Let $X_n = 1$ if $\sum_{i=1}^{n-1} X_i$ is odd and $X_n = 0$ otherwise. Let $n \ge 3$.

(a) Show that X_i and X_j are independent for $i \ne j$, $i, j \in \{1, 2, \ldots, n\}$.

(b) Find $H(X_i, X_j)$ for $i \ne j$.

(c) Find $H(X_1, X_2, \ldots, X_n)$. Is this equal to $nH(X_1)$?

(Assignment-3.5#337)

Functions of a stochastic process

(a) Consider a stationary stochastic process $X_1, X_2, \ldots, X_n$, and let $Y_1, Y_2, \ldots, Y_n$ be defined by

$$Y_i = \phi(X_i), \quad i = 1, 2, \ldots \quad (4.97)$$

for some function ϕ. Prove that

$$H(Y) \leq H(X). \quad (4.98)$$

(b) What is the relationship between the entropy rates $\mathcal{H}(\mathcal{Z})$ and $\mathcal{H}(\mathcal{X})$ if

$$Z_i = \psi(X_i, X_{i+1}), \quad i = 1, 2, \ldots \quad (4.99)$$

for some function ψ?

(Assignment-3.6#338)

Markov chain transitions

$$P = [P_{ij}] = \begin{bmatrix} \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{2} \end{bmatrix}.$$

Let X_1 be distributed uniformly over the states $\{0, 1, 2\}$. Let $\{X_i\}_{i=1}^{\infty}$ be a Markov chain with transition matrix P; thus, $P(X_{n+1} = j | X_n = i) = P_{ij}, i, j \in \{0, 1, 2\}$.

(a) Is $\{X_n\}$ stationary?

(b) Find $\lim_{n\to\infty} \frac{1}{n} H(X_1, \ldots, X_n)$.

Now consider the derived process $Z_1, Z_2, \ldots, Z_n$, where

$$Z_1 = X_1$$
$$Z_i = X_i - X_{i-1} \pmod 3, \quad i = 2, \ldots, n.$$

Thus, Z^n encodes the transitions, not the states.

(c) Find $H(Z_1, Z_2, \ldots, Z_n)$.

(d) Find $H(Z_n)$ and $H(X_n)$ for $n \geq 2$.

(e) Find $H(Z_n|Z_{n-1})$ for $n \geq 2$.

(f) Are Z_{n-1} and Z_n independent for $n \geq 2$?

(Assignment-3.7#339)

7. Huffman 20 questions. Consider a set of n objects. Let $X_i = 1$ or 0 accordingly as the ith object is good or defective. Let $X_1, X_2, \ldots, X_n$ be independent with $\Pr\{X_i = 1\} = p_i$; and $p_1 > p_2 > \cdots > p_n > \frac{1}{2}$. We are asked to determine the set of all defective objects. Any yes-no question you can think of is admissible.

(a) Give a good lower bound on the minimum average number of questions required.

(b) If the longest sequence of questions is required by nature's answers to our questions, what (in words) is the last question we should ask? What two sets are we distinguishing with this question? Assume a compact (minimum average length) sequence of questions.

(c) Give an upper bound (within one question) on the minimum average number of questions required.

(Assignment-3.8#340)

8. Shannon code. Consider the following method for generating a code for a random variable X that takes on m values $\{1, 2, \ldots, m\}$ with probabilities $p_1, p_2, \ldots, p_m$. Assume that the probabilities are ordered so that $p_1 \geq p_2 \geq \cdots \geq p_m$. Define

$$F_i = \sum_{k=1}^{i-1} p_k. \quad (5.148)$$

the sum of the probabilities of all symbols less than i. Then the codeword for i is the number $F_i \in [0, 1]$ rounded off to l_i bits, where $l_i = \lceil -\log p_i \rceil$.

(a) Show that the code constructed by this process is prefix-free and that the average length satisfies

$$H(X) \leq L < H(X) + 1. \quad (5.149)$$

(b) Construct the code for the probability distribution $(0.5, 0.25, 0.125, 0.125)$.

(Assignment-3.9#341)

9. Optimal codes for dyadic distributions. For a Huffman code tree, define the probability of a node as the sum of the probabilities of all the leaves under that node. Let the random variable X be drawn from a dyadic distribution [i.e., $p(x) = 2^{-i}$ for some i, for all $x \in \mathcal{X}$]. Now consider a binary Huffman code for this distribution.

(a) Argue that for any node in the tree, the probability of the left child is equal to the probability of the right child.

(b) Let $X_1, X_2, \ldots, X_n$ be drawn i.i.d. $\sim p(x)$. Using the Huffman code for $p(x)$, we map $X_1, X_2, \ldots, X_n$ to a sequence of bits $Y_1, Y_2, \ldots, Y_{L(X_1,X_2,\ldots,X_n)}$. (The length of this sequence will depend on the outcome $X_1, X_2, \ldots, X_n$). Use part (a) to argue that the sequence $Y_1, Y_2, \ldots$ forms a sequence of fair coin flips [i.e., that $\Pr\{Y_i = 0\} = \Pr\{Y_i = 1\} = \frac{1}{2}$, independent of $Y_1, Y_2, \ldots, Y_{i-1}$]. Thus, the entropy rate of the coded sequence is 1 bit per symbol.

(c) Give a heuristic argument why the encoded sequence of bits for any code that achieves the entropy bound cannot be compressible and therefore should have an entropy rate of 1 bit per symbol.

(Assignment-3.10#342)

Let P and Q be two PMFs on $\mathcal{A}$. Suppose P is the true PMF, but the designer assumes Q and encodes with lengths given by $\lceil -\log_2 Q(a) \rceil$ bits for the letter $a \in \mathcal{A}$. Give a bound on the excess in entropy for this mismatched compression, and write it in terms of relative entropy. How would your answer change for groups of n letters? In the limit?

(Assignment-4.1#343)

Consider a discrete memoryless source (DMS) on $\mathcal{A} = \{0, 1\}$ whose marginal is such that $\Pr\{X_1 = 1\} = 0.75$ or $\Pr\{X_1 = 1\} = 0.4$. The encoder and the decoder do not know which of these sources is the true one. What is $\lim_{n\to\infty} \frac{1}{n} r(u, \epsilon)$ for this 'uncertain' source?

(Assignment-4.2#344)

Show that in a sequence of $2k$ independent tosses of an unbiased coin, the probability of getting exactly k heads goes to 0 as $k \to \infty$.

(Assignment-4.3#345)

Yet, show that the probability of getting exactly k heads in $2k$ independent tosses of the unbiased coin is at least $1/(2k + 1)$ by showing that

$$\binom{2k}{k} \geq \binom{2k}{l} \quad \text{for any } l = 0, 1, \ldots, 2k.$$

[Hint: Prove that $a!b! \geq (a + b)!2^{-(a+b)}$, and use it.]

(Assignment-4.4#346)

Let τ and $\hat{\tau}$ be types that belong to $\mathcal{T}_n$. Let $A_n(\tau)$ be the set of sequences of type τ. Similarly $A_n(\hat{\tau})$. Consider the DMS with marginal τ on $\mathcal{A}$. Denote its n-letter PMF τ_n on $\mathcal{A}^n$. Show that $\tau_n(A_n(\tau)) \geq \tau_n(A_n(\hat{\tau}))$. [Hint: Write out the multinomial probabilities and use the previous problem's hint.]

(Assignment-4.5#347)

We proved that the upper bound for the number of types of sequences in $\mathcal{A}^n$ is $(n + 1)^{|\mathcal{A}|}$. Show that the exact number is

$$\binom{n + |\mathcal{A}| - 1}{|\mathcal{A}| - 1}.$$

(Assignment-4.6#348)

Use Stirling's approximation $k! \approx \sqrt{2\pi k}k^k e^{-k}$ and get an approximation for $\log|A_n(\tau)|$.

(Assignment-4.7#349)

Doubly stochastic matrices. An $n \times n$ matrix $P = [P_{ij}]$ is said to be *doubly stochastic* if $P_{ij} \geq 0$ and $\sum_j P_{ij} = 1$ for all i and $\sum_i P_{ij} = 1$ for all j. An $n \times n$ matrix P is said to be a *permutation matrix* if it is doubly stochastic and there is precisely one $P_{ij} = 1$ in each row and each column. It can be shown that every doubly stochastic matrix can be written as the convex combination of permutation matrices.

(a) Let $\mathbf{a} = (a_1, a_2, \ldots, a_n)$, $a_i \geq 0$, $\sum a_i = 1$, be a probability vector. Let $\mathbf{b} = \mathbf{a}P$, where P is doubly stochastic. Show that $\mathbf{b}$ is a probability vector and that $H(b_1, b_2, \ldots, b_n) \geq H(a_1, a_2, \ldots, a_n)$. Thus, stochastic mixing increases entropy.

(b) Show that a stationary distribution μ for a doubly stochastic matrix P is the uniform distribution.

(c) Conversely, prove that if the uniform distribution is a stationary distribution for a Markov transition matrix P, then P is doubly stochastic.

(Assignment-4.8#350)

Let $(X_i)_{i=-\infty}^{\infty}$ be a stationary stochastic process. Prove that

$$H(X_0|X_{-1}, X_{-2}, \ldots, X_{-n}) = H(X_0|X_1, X_2, \ldots, X_n).$$

In other words, the present has a conditional entropy given the past equal to the conditional entropy given the future. This is true even though it is quite easy to concoct stationary random processes for which the flow into the future looks quite different from the flow into the past. That is, one can determine the direction of time by looking at a sample function of the process. Nevertheless, given the present state, the conditional uncertainty of the next symbol in the future is equal to the conditional uncertainty of the previous symbol in the past.

(Assignment-4.9#351)

9. Error exponent for universal codes. A universal source code of rate R achieves a probability of error $P_e^{(n)} \doteq e^{-nD(P^*\|Q)}$, where Q is the true distribution and P^* achieves $\min_{P:H(P)\geq R} D(P\|Q)$.

(a) Find P^* in terms of Q and R.

(b) Now let X be binary. Find the region of source probabilities $Q(x), x \in \{0,1\}$, for which rate R is sufficient for the universal source code to achieve $P_e^{(n)} \to 0$.

(Assignment-4.10#352)

Give an example of pair of distributions P_1 and P_2 such that $D(P_1\|P_2) \neq D(P_2\|P_1)$.

(Assignment-5.1#353)

Let $P^{(0)}$ and $P^{(1)}$ be two PMFs on $\mathcal{A}$ such that for every $a \in \mathcal{A}$, both $P^{(0)}(a) > 0$ and $P^{(1)}(a) > 0$. $X \sim P^{(0)}$ under H_0 and $X \sim P^{(1)}$ under H_1. Consider the likelihood ratio test with threshold 0, i.e.,

$$D_1 = \left\{ a \in \mathcal{A} \mid \frac{P^{(1)}(a)}{P^{(0)}(a)} \geq 1 \right\}.$$

Consider a decision region C_1 such that $P^{(0)}(C_1) \leq P^{(0)}(D_1)$. Argue that

$$P^{(1)}(C_1^c) \geq P^{(1)}(D_1^c).$$

(Assignment-5.2#354)

Consider the following two hypotheses with alphabet $\mathbb{R}$:

$$\begin{aligned} H_0 &: X = \mu_0 + Z \\ H_1 &: X = \mu_1 + Z \end{aligned}$$

where Z has the Normal distribution with zero mean and unit variance, and $\mu_1 > \mu_0$. Find the likelihood ratio (of H_1 w.r.t. H_0), the log-likelihood ratio, and the relative entropy. Discuss some interesting properties of this relative entropy.

(Assignment-5.3#355)

Consider two zero-means Gaussians $\mathcal{N}(0, \sigma_1^2)$ and $\mathcal{N}(0, \sigma_2^2)$ of different variances. Compute the relative entropy $D(\mathcal{N}(0, \sigma_1^2) \| \mathcal{N}(0, \sigma_2^2))$.

(Assignment-5.4#356)

Let X and Y be Poisson random variables with means λ and μ respectively. Compute the relative entropy $D(P_X \| P_Y)$.

(Assignment-5.5#357)

In 100 throws of an unbiased die, the observed average number of dots per throw was 5. Find the PMF P^* in whose neighbourhood the empirical distribution will lie with high probability. (E.g.,, 80 sixers, 20 singles, and none of the others? 33 sixers, 33 fours, 34 fives, and none of the others?) You may need the help of a computer to find the exact P^*. Those who have played bridge may have some intuition on questions of this nature.

(Assignment-5.6#358)

Suppose that $X_1, X_2, \ldots, X_n$ are iid with mean 0. Consider $\Pr\{\frac{1}{n}\sum_{i=1}^{n} X_i > a\}$ for $a > 0$. In order to get the Chebyshev inequality, you squared both sides and applied Markov inequality. Instead, assume X_1 has exponential moments, apply the function $e^{t(\cdot)}$ to both sides with $t \geq 0$, and apply Markov inequality. Optimising the upper bound over $t \geq 0$, can you show that the probability decays exponentially fast to zero?

(Assignment-5.7#359)

Suppose that $\{P_{X|\theta}, \theta \in \Theta\}$ is a family of PMFs on $\mathcal{A}$. Let $Y : \mathcal{A} \to \mathcal{B}$, and let there be functions $g_\theta : \mathcal{B} \to \mathbb{R}_+$ and $h : \mathcal{A} \to \mathbb{R}_+$ such that

$$P_{X|\theta}(a) = g_\theta(T(a))h(a), \quad \text{for all } a \in \mathcal{A} \text{ and } \theta \in \Theta.$$

Show that T is a sufficient statistic for θ.

(Assignment-5.8#360)

Suppose that there are two coins with biases p and r respectively. One of these coins was picked and tossed n times. Let $X_1, X_2, \ldots, X_n$ be the outcomes of these tosses with $X_i = 1$ if the ith toss was a "Head" and 0 otherwise. To save space, your lab mate stored $Y = \sum_{i=1}^{n} X_i$ and deleted the exact sequence of toss outcomes. Identify the loss in relative entropy and interpret your answer in terms of being able to decide on the bias.

(Assignment-6.1#361)

Given an example of n, P_{X^n}, Q_{X^n} such that $D(P_{X^n}\|Q_{X^n}) \neq \sum_{i=1}^n D(P_{X_i}\|Q_{X_i})$.

(Assignment-6.2#362)

Show the following:

(a) $I(X;Y|Z) = D(P_{X|YZ}\|P_{X|Z}|P_{YZ})$.

(b) $I(X_1, X_2, \ldots, X_n; Y) = I(X_1;Y) + \sum_{i=2}^n I(X_i;Y|X_1, X_2, \ldots, X_{i-1})$.

(Assignment-6.3#363)

Prove or disprove: $I(X;Y;Z) \geq I(X;Y)$.

(Assignment-6.4#364)

Measure of correlation. Let X_1 and X_2 be identically distributed but not necessarily independent. Let

$$\rho = 1 - \frac{H(X_2|X_1)}{H(X_2)}.$$

(a) Show that $\rho = \frac{I(X_1;X_2)}{H(X_2)}$.

(b) Show that $0 \leq \rho \leq 1$.

(c) When is $\rho = 0$?

(d) When is $\rho = 1$?

(Assignment-6.5#365)

5. Example of joint entropy. Let $p(x, y)$ be given by

$y \setminus x$	0	1
0	1/3	1/3
1	0	1/3

Find:

(a) $H(X), H(Y)$.

(b) $H(X|Y), H(Y|X)$.

(c) $H(X, Y)$.

(d) $H(Y) - H(Y|X)$.

(e) $I(X; Y)$.

(f) Draw a Venn diagram for the quantities in parts (a) through (e).

(Assignment-6.6#366)

6. Data processing. Let $X_1 \to X_2 \to X_3 \to \cdots \to X_n$ form a Markov chain in this order; that is, let

$$p(x_1, x_2, \ldots, x_n) = p(x_1)p(x_2|x_1) \cdots p(x_n|x_{n-1}).$$

Reduce $I(X_1; X_2, \ldots, X_n)$ to its simplest form.

(Assignment-6.7#367)

7. Random questions. One wishes to identify a random object $X \sim r(x)$. A question $Q \sim q(q)$ is asked at random according to $r(q)$. This results in a deterministic answer $A = A(x, q) \in \{a_1, a_2, \ldots\}$. Suppose that X and Q are independent. Then $I(X; Q, A)$ is the uncertainty in X removed by the question-answer (Q, A).

(a) Show that $I(X; Q, A) = H(A|Q)$. Interpret.

(b) Now suppose that two i.i.d. questions $Q_1, Q_2 \sim r(q)$ are asked, eliciting answers A_1 and A_2. Show that two questions are less valuable than twice a single question in the sense that $I(X; Q_1, A_1, Q_2, A_2) \leq 2I(X; Q_1, A_1)$.

(Assignment-6.8#368)

8. Inequalities. Which of the following inequalities are generally $\geq, =, \leq$? Label each with $\geq, =,$ or $\leq$.

(a) $H(5X)$ vs. $H(X)$

(b) $I(g(X);Y)$ vs. $I(X;Y)$

(c) $H(X_0|X_{-1})$ vs. $H(X_0|X_{-1}, X_1)$

(d) $\frac{H(X,Y)}{H(X)+H(Y)}$ vs. 1

(Assignment-6.9#369)

9. Mutual information of heads and tails

(a) Consider a fair coin flip. What is the mutual information between the top and bottom sides of the coin?

(b) A six-sided fair die is rolled. What is the mutual information between the top side and the front face (the side facing you)?

(Assignment-6.10#370)

10. Show that $D(P_\lambda \| P_0)$ is an increasing function of λ for $\lambda \in [0,1]$ where

$$P_\lambda(x) = \frac{P_1(x)^\lambda P_0(x)^{1-\lambda}}{\sum_{a\in\mathcal{A}} P_1(a)^\lambda P_0(a)^{1-\lambda}}.$$

(Assignment-7.1#371)

Prove the "joint" asymptotic equipartition property for an $A(n, \delta)$ that consists of sequences whose empirical and both marginal frequencies (normalized histograms) are within a small error around the true PMFs.

$P_{X^{(n)},Y^{(n)}}(X, Y)$ denotes the empirical joint distribution.

$P_{X^{(n)}}(X)$ denotes the empirical marginal of X's distribution.

$P_{Y^{(n)}}(Y)$ denotes the empirical marginal of Y's distribution.

$A(n, \delta)$ is defined as

$$\begin{aligned} A(n, \delta) = \Big\{(x, y) :& |P_{X^{(n)},Y^{(n)}}(x, y) - P_{XY}(x, y)| \leq \delta''' \\ & |P_{X^{(n)}}(x) - P_X(x)| \leq \delta' \\ & |P_{Y^{(n)}}(y) - P_Y(y)| \leq \delta'' \Big\} \end{aligned}$$

For some small errors $\delta', \delta'', \delta''' > 0$. $\delta = \max\{\delta', \delta'', \delta'''\}$.

$$\begin{aligned} &Pr\{|P_{X^{(n)},Y^{(n)}}(x, y) - P_{XY}(x, y)| \leq \delta'''\} = 1 \\ &Pr\{|P_{X^{(n)}}(x) - P_X(x)| \leq \delta'\} = 1 \\ &Pr\{|P_{Y^{(n)}}(y) - P_Y(y)| \leq \delta''\} = 1 \end{aligned}$$

for sequences in $A(n, \delta)$. Now,

$$\begin{aligned} &Pr\{|P_{X^{(n)}}(x) - P_X(x)| \leq \delta'\} = 1 \\ &Pr\{|\frac{N(x)}{n} - P_X(x)| \leq \delta'\} = 1 \\ &Pr\{|\frac{\sum_{i=1}^{n} \mathbb{I}_{X_i=x}}{n} - P_X(x)| \leq \delta'\} = 1 \end{aligned}$$

indicates n is sufficiently large enough for WLLN to kick-in for δ'. For a sequence of n i.i.d random variables $(X_1, \dots, X_n)$,

$$Pr\{\Big|\frac{1}{n}\sum_{i=1}^{n} \log\frac{1}{P(X_i)} - H(X)\Big| \leq \epsilon'\} = 1$$

by WLLN, where ϵ' is a small value and $\epsilon' > 0$.

Similarly, we have

$$Pr\{\Big|\frac{1}{n}\sum_{i=1}^{n} \log\frac{1}{P(Y_i)} - H(Y)\Big| \leq \epsilon''\} = 1$$

Now,

$$Pr\{|P_{X^{(n)},Y^{(n)}}(x,y) - P_{XY}(x,y)| \leq \delta'''\} = 1$$
$$\therefore Pr\{\Big|\frac{1}{n}\sum_{i=1}^{n}\log\frac{1}{P_{XY}(X_i,Y_i)} - H(XY)\Big| \leq \epsilon'''\} = 1$$

Let $\epsilon = \max\{\epsilon', \epsilon'', \epsilon'''\}$. Then,

$$Pr\Big\{\Big|\frac{1}{n}\sum_{i=1}^{n}\log\frac{1}{P(X_i)} - H(X)\Big| \leq \epsilon\Big\}$$
$$Pr\Big\{\Big|\frac{1}{n}\sum_{i=1}^{n}\log\frac{1}{P(Y_i)} - H(Y)\Big| \leq \epsilon\Big\}$$
$$Pr\Big\{\Big|\frac{1}{n}\sum_{i=1}^{n}\log\frac{1}{P_{XY}(X_i,Y_i)} - H(XY)\Big| \leq \epsilon\Big\}$$

Therefore,

$$\frac{1}{n}\log P_{X^n,Y^n}(X^n,Y^n) \to H(XY)$$
$$\frac{1}{n}\log P_{X^n}(X^n) \to H(X)$$
$$\frac{1}{n}\log P_{Y^n}(Y^n) \to H(Y)$$

in probability, for sequences in $A(n,\delta)$.

(Assignment-7.2#372)

Show that capacity does not change if $P_e^{(n)}(c)$, the average probability of error, is replaced by $\overline{P}_e^{(n)}(c)$, the maximum probability of error.

$$P_e^{(n)}(w) = Pr\{\phi_n(Y^n) \neq w | f_n(w)\}$$
$$P_e^{(n)}(w) = \frac{1}{M_n}\sum_{i=1}^{M_n} P_e^{(n)}(w=i)$$

Let C_{avg} be the capacity achieved with $P_e^{(n)} \leq \epsilon$.

Let $C_{\max}$ be the capacity achieved with $\overline{P}_e^{(n)} \leq \epsilon$, where $\overline{P}_e^{(n)}$ is the maximum probability of error. Our objective is to show $C_{\text{avg}} = C_{\max}$.

For a sequence of codes (n, M_n) such that $P_e^{(n)} \leq \epsilon/2$,

$$\frac{\log(M_n)}{n} \geq R - \frac{\eta}{2}$$

Prune the codebook to remove all codewords with $P_e^{(n)}(w) > \epsilon$. This implies the number of codewords left $\geq M_n/2$;

$$P_e^{(n)} \leq \overline{P}_e^{(n)} \leq \epsilon$$

$$\begin{aligned} \frac{\log \# \text{ of codewords}}{n} &\geq \frac{\log \frac{M_n}{2}}{n} \\ &= \frac{\log M_n}{n} - \frac{1}{n} \\ &= R - \frac{\eta}{2} - \frac{1}{n} \\ &\geq_n R - \eta \end{aligned}$$

$$\begin{aligned} C_{\text{avg}} &= \sup\{R : \exists (n, M_n) \text{ such that } P_e^{(n)} \leq_n \epsilon\} \\ C_{\max} &= \sup\{R : \exists (n, M_n) \text{ such that } \overline{P}_e^{(n)} \leq_n \epsilon\} \end{aligned}$$

$$\boxed{C_{\text{avg}} = C_{\max}}$$

as the same rate R is achieved fdor both $P_e^{(n)} \leq \epsilon$ and $\overline{P}_e^{(n)} \leq \epsilon$ conditions.

(Assignment-7.3#373)

Can you apply the source-channel separation principle for a two-terminal system with feedback? In other words, is a stationary and ergodic source with $H > C$ transmissible over a DMC with perfect feedback?

By two terminal system, one means a communication channel with two terminals (end-points), one at the transmitter and one at the decoder end. The question is asking us to simply consider a DMC with feedback.

The answer is yes; we can use source-channel separation principle to a two-terminal system with feedback.

Note that $C_F = C$. When $H > C \implies H > C_F$; therefore a stationary and ergodic sourrce with $H > C$ is not transmissible over a DMC with perfect feedback. (Proof is the same as converse of source-channel separation theorem using Fano's inequality).

(Assignment-7.4#374)

Find the capacity of the noisy typewriter with five input symbols.

(Assignment-7.5#375)

Find a code for the noisy typewriter with five input symbols such that the probability of error is zero, and the rate is strictly larger than 1 bit per transmission.

(Assignment-7.6#376)

Consider a channel whose input alphabet is 0, 17, i.e., 7 bits. The channel is such that either there is no error, or exactly one of the seven bits is flipped. These eight possibilities have equal probabilities. What is the capacity of this channel? Can you come up with a code operating at that capacity? (If you can, find your code's probability of error).

(Assignment-7.7#377)

uppose F0 = 1, F1 = 2, and define Fn = Fn−1 + Fn−2 for n 2. Show that Fn grows exponentially with n, i.e., there is a > 1 such that n−1 log F (n) → log , so that Fn n for large n. What if Fn = k i=1 aiFn−i with some initial condition?

(Assignment-7.8#378)

Consider the following telegraph channel where dots and dashes are used as input symbols. The channel is noiseless. Every dot lasts one unit of time and every dash lasts two units of time. Get a recurrence relation for the total number of strings that last n units of time. Hence deduce the capacity of this channel in bits per unit time.

(Assignment-7.9#379)

Consider the binary symmetric channel with cross-over probability > 0. For an arbitrary code word, what is the support set of the output string? If your code must have zero probability of error, what is the (zero-error) capacity?

(Assignment-7.10#380)

Consider a binary input binary output channel with $W(0|0) = 1$ and $W(1|1) =$, where $0 < \; < 1$. This is called a Z-channel for reasons that will be obvious when you draw the channel picture. Call the capacity of this channel C( ). Show that lim ↓0 C( )/ = e−1.

(Assignment-8.1#381)

1. Prove the following for the $A(n,\delta)$ defined in the continuum case.

(a) $\lim_{n\to\infty} P\{A(n,\delta)\} = 1$.

(b) $\text{Vol}(A(n,\delta)) \le 2^{nh(X)+n\delta}$ for all n.

(c) $\text{Vol}(A(n,\delta)) \ge (1-\epsilon)\cdot 2^{nh(X)-n\delta}$.

(Assignment-8.2#382)

Evaluate the differential entropy $h(X) = -\int f \ln f$ for the following:

(a) The exponential density, $f(x) = \lambda e^{-\lambda x}, x \ge 0$.

(b) The Laplace density, $f(x) = \frac{\lambda}{2} e^{-\lambda|x|}$.

(c) The sum of X_1 and X_2, where X_1 and X_2 are independent normal random variables with means μ_i and variances σ_i^2, $i = 1, 2$.

(Assignment-8.3#383)

Gaussian mutual information. Suppose that (X, Y, Z) are jointly Gaussian and that $X \to Y \to Z$ forms a Markov chain. Let X and Y have correlation coefficient ρ_1 and let Y and Z have correlation coefficient ρ_2. Find $I(X;Z)$.

(Assignment-8.4#384)

Shape of the typical set. Let X_i be i.i.d. $\sim f(x)$, where

$$f(x) = ce^{-x^4}.$$

Let $h = -\int f \ln f$. Describe the shape (or form) or the typical set

$$A_\epsilon^{(n)} = \{x^n \in \mathbb{R}^n : f(x^n) \in 2^{-n(h\pm\epsilon)}\}.$$

(Assignment-8.5#385)

5. Let $\mathcal{P}'$ and $\mathcal{Q}'$ be refinements of the partitions $\mathcal{P}$ and $\mathcal{Q}$, respectively. Let X and Y be two real-valued random variables with joint distribution $P_{X,Y}$. Show that

$$I(X;Y) \geq I(X|_{\mathcal{P}};Y|_{\mathcal{Q}}).$$

(Assignment-8.6#386)

6. Prove the following saddle-point inequality:

$$I(X;X+Z_G) \leq I(X_G;X_G+Z_G) \leq I(X_G;X_G+Z),$$

where $X_G \sim \mathcal{N}(0,P)$ and X is a random variable with mean 0 and variance P, while $Z_G \sim \mathcal{N}(0,\sigma^2)$ and Z is a random variable with mean 0 and variance σ^2. (Assume that in any sum, the constituent random variables are independent of each other). While the first inequality is what you used to show the capacity of the AWGN channel, what conclusion can you draw from the second inequality?

(Assignment-8.7#387)

7. Let us try to get a saddle-point inequality for the discrete case. Let the input and output alphabets be of finite size. Fix $P_{Y|X}$. Let P_{X^*} maximise $I(X;Y)$ and P_{Y^*} be the corresponding marginal of Y. Which of the inequalities in the following statement is true? For any P_X, and any Q,

$$D(P_{Y|X}\|P_{Y^*}|P_X) \leq D(P_{Y|X}\|P_{Y^*}|P_{X^*}) \leq D(P_{Y|X}\|Q|P_{X^*}).$$

(Assignment-8.8#388)

8. Consider an ideal gas made of noninteracting (i.e., they do not collide) particles in a horizontal container of finite length. Each particle is of mass m and executes only horizontal motion. Reflections at the two boundaries are elastic so that at these locations the velocity of the particle gets reversed without a change in magnitude. Suppose that a particle's velocity V is distributed according to the density function p satisfying

- $E_p[V] = 0$,
- $E_p[\frac{1}{2}mV^2] = \frac{1}{2}m\sigma^2$; (temperature is a measure of average kinetic energy).

What is the p satisfying the above conditions that maximises the differential entropy?

(Assignment-8.9#389)

9. Let P_X be a PMF on A and $f : A \times B \to \mathbb{R}_+$ a function such that

$$\mathbb{E}[f(X, y)] = 1, \quad \forall y \in B,$$

where the expectation is with respect to P_X. Suppose that M codewords $x_1, \ldots, x_M$ are picked, independently of each other, each according to PMF P_X. Let the first codeword be transmitted over the channel $P_{Y|X}$, and let y be received. Let $A_i = \{f(x_i, y) > \beta\}$. Show that

$$P\left\{\bigcup_{i=2}^{M} A_i \mid \text{Message 1}\right\} \leq \frac{M}{\beta}.$$

(Assignment-8.10#390)

(a) For a code transmitting reliably at R bits per second on a channel of passband bandwidth W Hz, average power constraint $\bar{P}$ Watts, what is the energy per bit?

(b) What is the least energy per bit that is needed for reliable transmission?

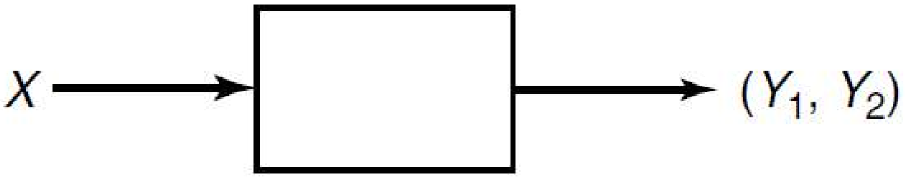
(c) What is bandwidth W needed for a fixed $\bar{P}$ to attain this minimum energy per bit?

(Assignment 9.1#391)

Two look gaussian channel: Consider the ordinary gaussian channel with two correlated looks at X, that is, $Y = (Y_1, Y_2)$, where

$$Y_1 = X + Z_1$$
$$Y_2 = X + Z_2$$

with a power constraint P on X, and $(Z_1, Z_2) \sim \mathcal{N}(0, K)$, where



$$\mathcal{K} = \begin{bmatrix} N & N\rho \\ N\rho & N \end{bmatrix}$$

Find the capacity C for

1. $\rho = 1$
2. $\rho = 0$
3. $\rho = -1$

Give correlated looks,

$$Y_1 = X + Z_1$$
$$Y_2 = X + Z_2$$

$$\mathbb{E}\|X^2\| \leq P$$

$$(Z_1, Z_2) \sim \mathcal{N}(0, \mathcal{K})$$

where

$$\mathcal{K} = \begin{bmatrix} N & N\rho \\ N\rho & N \end{bmatrix}$$

Objective is to find the capacity C of the channel.

$$\begin{aligned} C &= \max I(X; Y_1, Y_2) \\ &= h(Y_1, Y_2) - h(Y_1, Y_2|X) \\ &= h(Y_1, Y_2) - h(Z_1, Z_2) \end{aligned}$$

as uncertainty to (Y_1, Y_2) if we already know X comes only from Z_1 and Z_2 both of which are independent of X.

We know that

$$h(Z_1, Z_2) = \frac{1}{2}\log_2(2\pi e^2|\mathcal{K}|)$$

substituting the determinant of $\mathcal{K}$,

$$h(Z_1, Z_2) = \frac{1}{2}\log_2\left(2\pi e^2 N^2(1-\rho^2)\right)$$

The distribution of X that maximizes mutual information $I(X; Y_1, Y_2)$ will in turn maximize the channel capacity C. Such a distribution would be the gaussian distribution $X \sim \mathcal{N}(0, P)$.

Now we look at the gaussian joint distribution (Y_1, Y_2): $\mathbb{E}(Y_1) = \mathbb{E}(Y_2) = 0$.

$$\begin{aligned}\text{var}(Y_i) &= \text{var}(X_i + Z_i)\\ &= \text{var}(X_i) + \text{var}(Z_i)\\ &= P + N\end{aligned}$$

$$\begin{aligned}\text{Cov}(Y_1, Y_2) &= \mathbb{E}\left(Y_1 - \mathbb{E}\left[Y_1\right)\left(Y_2 - \mathbb{E}Y_2\right)\right]\\ &= \mathbb{E}\left[Y_1 Y_2\right]\\ &= \mathbb{E}\left[(X+Z_1)(X+Z_2)\right]\\ &= \mathbb{E}\left[X^2 + Z_1 Z_2 + X(Z_1+Z_2)\right]\\ &= \mathbb{E}\left[X^2\right] + \mathbb{E}\left[Z_1 Z_2\right]\\ &\text{other terms cancel as } X \text{ and } Z_1\text{,}Z_2 \text{ are independent.}\\ &= P + N\rho\\ \mathcal{K}_{Y_1,Y_2} &= \begin{bmatrix} P+N & P+N\rho \\ P+N\rho & P+N \end{bmatrix}\end{aligned}$$

$$\begin{aligned}|\mathcal{K}_{Y_1,Y_2}| &= (N+P)^2 - (P+N\rho))^2\\ &= \left(N^2 + P^2 + 2NP\right) + \left(P^2 + N^2\rho^2 + 2N\rho\right)\\ &= N^2(1-\rho^2) + 2NP(1-\rho)\end{aligned}$$

$$\begin{aligned}h(Y_1, Y_2) &= \frac{1}{2}\log_2\left(2\pi e^2|\mathcal{K}_{Y_1,Y_2}|\right)\\ &= \frac{1}{2}\log_2\left(2\pi e^2\left[N^2(1-\rho^2) + 2NP(1-\rho)\right]\right)\\ C &= h(Y_1, Y_2) - h(Z_1, Z_2)\\ &= \frac{1}{2}\log_2\left(2\pi e^2\left[N^2(1-\rho^2) + 2NP(1-\rho)\right]\right) - \frac{1}{2}\log_2\left(2\pi e^2 N^2(1-\rho^2)\right)\\ &= \frac{1}{2}\log_2\left(1 + \frac{2P}{N(1+\rho)}\right)\end{aligned}$$

1. $\rho = 1$

$$C = \frac{1}{2}\log_2\left(1 + \frac{P}{N}\right)$$

2. $\rho = 0$

$$C = \frac{1}{2}\log_2\left(1+\frac{2P}{N}\right)$$

3. $\rho = -1$

$$C = \frac{1}{2}\log_2\left(1+\frac{P}{N(1-1)}\right) = \infty$$

(Assignment 9.2#392)

Exponential noise channels: $Y_i = X_i + Z_i$, where Z_i is i.i.d exponentially distributed noise with mean μ. Assume that we have a mean constraint on the signal (i.e., $\mathbb{E}X_i \leq \lambda$). Show that the capacity of such a channel is

$$C = \log\left(1+\frac{\lambda}{\mu}\right).$$

Exponential noise channels

$$Y_i = X_i + Z_i$$

Z_i → exponentially distributed i.i.d exponential with mean μ.

$$\mathbb{E}X_i \leq \lambda$$

To show:

$$C = \log\left(1+\frac{\lambda}{\mu}\right)$$

We know that

$$\begin{aligned} C &= \max_{\mathbb{E}X\leq\lambda}\ I(X;Y) \\ &= \max_{\mathbb{E}X\leq\lambda} h(Y) - h(Y|X) \\ &= \max_{\mathbb{E}X\leq\lambda} h(Y) - h(X+Z \mid X) \\ &= \max_{\mathbb{E}X\leq\lambda} h(Y) - h(Z) \end{aligned}$$

We now try to find $h(Z)$. We'll use nats as the units for entropies.

$$f_Z(Z) = \frac{1}{\mu}e^{-z/\mu}, \qquad z \geq 0$$

The differential entropy of Z is defined as:

$$h(Z) = -\int_0^\infty f_Z(z)\log f_Z(z)\,dz$$

Substitute the PDF:

$$h(Z) = -\int_0^\infty \frac{1}{\mu} e^{-z/\mu} \log\left(\frac{1}{\mu} e^{-z/\mu}\right) dz$$

Use the logarithmic identity:

$$\log\left(\frac{1}{\mu} e^{-z/\mu}\right) = \log\left(\frac{1}{\mu}\right) - \frac{z}{\mu}$$

Then:

$$h(Z) = -\int_0^\infty \frac{1}{\mu} e^{-z/\mu} \left[\log\left(\frac{1}{\mu}\right) - \frac{z}{\mu}\right] dz$$

Break into two integrals:

$$h(Z) = -\log\left(\frac{1}{\mu}\right) \int_0^\infty \frac{1}{\mu} e^{-z/\mu} dz \quad + \quad \int_0^\infty \frac{1}{\mu^2} z e^{-z/\mu} dz$$

Compute the first integral:

$$\int_0^\infty \frac{1}{\mu} e^{-z/\mu} dz = 1$$

Compute the second integral (expected value of Z, use "ILATE" rule of integration of product of two functions):

$$\int_0^\infty \frac{1}{\mu^2} z e^{-z/\mu} dz = \mu$$

So:

$$h(Z) = -\log\left(\frac{1}{\mu}\right) + \mu \cdot \frac{1}{\mu} = \log(\mu) + 1$$

$$\boxed{h(Z) = 1 + \log \mu} \quad \text{(in nats)}$$

Now using $Y = X + Z$, $\mathbb{E}Y \leq \lambda + \mu$.

Fact: Given the mean constraint, entropy is maximized by exponential distribution. So we have

$$\max_{\mathbb{E}Y \leq \lambda + \mu} h(Y) = 1 + \ln(\lambda + \mu)$$

and

$$C = \ln(1 + \lambda/\mu)$$

But the solution isn't complete. We have to know what is the distribution of Y. In gaussian case, if X and Z are gaussian, Y is also a gaussian, but in exponential case, this need not hold. To get the distribution of Y, we use the characteristic functions of the distributions. Characteristic functions of a random variable X is

$$\Psi_X(t) = \mathbb{E}\left[e^{itX}\right]$$

Characteristic function of exponential function would be

$$\Psi_X(t) = \frac{1}{1 - j\mu t}$$

For X and Y being two random variables,

$$\Psi_{X+Y}(t) = \Psi_X(t) + \Psi_Y(t)$$

$Y = X + Z$, so $\Psi_{X+Z} = \Psi_Y(t) = \Psi_X(t) + \Psi_Y(t)$.

We have

$$\Psi_Y(t) = \frac{1 - j\mu t}{1 - j(\mu + \lambda)t}$$

interpreted as shifted point mass, where $Y = 0$, with probability p and $Y = \exp(\lambda + \mu)$ with probability $1 - p$.

Note: Obtaining the characteristic function of exponential function:

Let $Z \sim \text{Exp}(\mu)$ with probability density function:

$$f_Z(z) = \frac{1}{\mu} e^{-z/\mu}, \quad z \geq 0$$

The characteristic function $\varphi_Z(t)$ is defined as:

$$\varphi_Z(t) = \mathbb{E}[e^{itZ}] = \int_{-\infty}^{\infty} e^{itz} f_Z(z)\, dz$$

Since $f_Z(z) = 0$ for $z < 0$, we get:

$$\varphi_Z(t) = \int_0^{\infty} e^{itz} \cdot \frac{1}{\mu} e^{-z/\mu}\, dz$$

Combine the exponentials:

$$\varphi_Z(t) = \int_0^{\infty} \frac{1}{\mu} e^{-z\left(\frac{1}{\mu} - it\right)}\, dz$$

Let $a = \frac{1}{\mu} - it$. Since $\text{Re}(a) = \frac{1}{\mu} > 0$, the integral converges:

$$\varphi_Z(t) = \frac{1}{\mu} \int_0^{\infty} e^{-az}\, dz = \frac{1}{\mu} \cdot \frac{1}{a} = \frac{1}{1 - i\mu t}$$

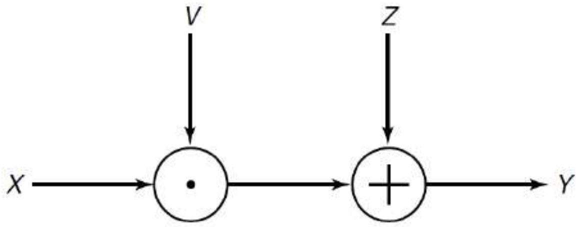
$$\boxed{\varphi_Z(t) = \frac{1}{1 - i\mu t}}$$

(Assignment 9.3#393)

Fading channel: Consider an additive noise fading channel

$$Y = XV + Z,$$

where 'Z is additive noise, V is a random variable representing fading, and Z and V are



independent of each other and of X. Argue that the knowledge of the fading factor V improves the capacity by showing that

$$I(X;Y|V) \geq I(X;Y)$$

Given $Z \perp\!\!\!\perp V \perp\!\!\!\perp X$.

$$\begin{aligned} I(X;Y,V) &= I(X;Y) + I(X;V|Y) \\ &= I(X;V) + I(X;Y|V) \end{aligned}$$

But we know that $IX(;V) = 0$ as $X \perp\!\!\!\perp V$. Therefore,

$$I(X;Y) + I(X;V|Y) = I(X;Y|V)$$

and $I(X;V|Y) \geq 0$ using the fact that mutual information is always non-negative. This means

$$I(X;Y) \leq I(X;Y|V)$$

(Assignment 9.4#394)

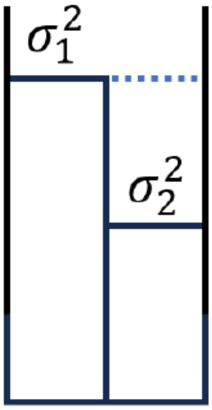
Parallel channels and water-filling: Consider a pair of gaussian channels

$$\begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix} = \begin{pmatrix} X_1 \\ X_2 \end{pmatrix} + \begin{pmatrix} Z_1 \\ Z_2 \end{pmatrix}$$

where

$$\begin{pmatrix} Z_1 \\ Z_2 \end{pmatrix} \sim N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{bmatrix} \sigma_1^2 & 0 \\ 0 & \sigma_2^2 \end{bmatrix} \right)$$

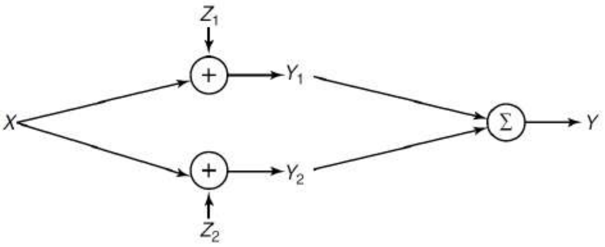
and there is a power constraint $\mathbb{E}\left(X_1^2 + X_2^2\right) \leq 2P$. Assume that $\sigma_1^2 \geq \sigma_2^2$. At what power does the channel stop behaving like a single channel with noise variance σ_2^2 and begin behaving like a pair of channels?



We use water-filling logic here. Initially for low values of power, the power will be allocated for channel 2. In this scenario, the pair of gaussian channels behave like a single channel as only channel 2 is being used here. It is when $P > \sigma_1^2 - \sigma_2^2$ that there is power allocation in both channels and the system starts behaving like a pair of channels.

(Assignment 9.5#395)

Multipath gaussian channel: Consider a gaussian noise channel with power constraint P, where the signal takes two different paths and the received noisy signals are added together at the antenna.



1. Find the capacity of this channel if Z_1 and Z_2 are jointly normal with covariance matrix

$$\mathcal{K}_Z = \begin{pmatrix} \sigma^2 & \rho\sigma^2 \\ \rho\sigma^2 & \sigma^2 \end{pmatrix}$$

2. What is the capacity for $\rho = 0$, $\rho = 1$ and $\rho = -1$?

$$
\begin{aligned}
Y_1 &= X_1 + Z_1 \\
Y_2 &= X_2 + Z_2 \\
Y &= Y_1 + Y_2 \\
&= X_1 + X_2 + Z_1 + Z_2 \\
&= 2X_1 + Z_1 + Z_2
\end{aligned}
$$

with $\mathbb{E}X^2 \leq P$. Since at the channel output end we're seeing $2X$ instead of X, the signal power would be $\mathbb{E}(2X)^2 = 4\mathbb{E}X = 4P$.

Capacity of this channel would be

$$C = \frac{1}{2}\log_2\left(1 + \frac{4P}{\mathbb{E}\left[Z_1 + Z_2\right]^2}\right)$$

$$
\begin{aligned}
\mathbb{E}\left[(Z_1 + Z_2)^2\right] &= \mathbb{E}Z_1^2 + \mathbb{E}Z_2^2 + 2\mathbb{E}Z_1Z_2 \\
&= \sigma^2 + \sigma^2 + 2\rho\sigma^2 \\
&= 2\sigma^2(1+\rho)
\end{aligned}
$$

$$
\begin{aligned}
C &= \frac{1}{2}\log_2\left(1 + \frac{4P}{\mathbb{E}\left[Z_1 + Z_2\right]^2}\right) \\
&= \frac{1}{2}\log_2\left(1 + \frac{4P}{2\sigma^2(1+\rho)}\right)
\end{aligned}
$$

Capacity when

- $\rho = 0$

$$C = \frac{1}{2}\log_2\left(1 + \frac{2P}{\sigma^2}\right)$$

- $\rho = 1$

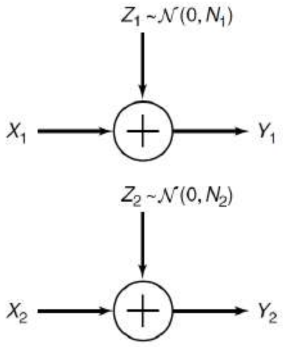
$$C = \frac{1}{2}\log_2\left(1 + \frac{P}{\sigma^2}\right)$$

- $\rho = -1$

$$C = \infty$$

(Assignment 9.6#396)

Parallel gaussian channels: Consider the following parallel gaussian channel: where $Z_1 \sim \mathcal{N}(0, N_1)$ and $Z_2 \sim \mathcal{N}(0, N_2)$ are independent gaussian random variables and $Y_i = X_i + Z_i$. We wish to allocate power to the two gaussian channels. Let β_1 and β_2 be fixed. Consider a total cost constraint $\beta_1 P_1 + \beta_2 P_2 \leq \beta$, where P_i is the power allocated to the i-th channel and β_i is the cost per unit power in that channel. Thus $P_1 \geq 0$ and $P_2 \geq 0$ can be chosen subject to the cost constraint β.



$Z_1 \sim \mathcal{N}(0, N_1)$
X_1
Y_1
$Z_2 \sim \mathcal{N}(0, N_2)$
X_2
Y_2

1. For what value of β does the channel stop acting like a single channel and start acting like a pair of channels?
2. Evaluate the capacity and find P_1 and P_2 that achieve capacity for $\beta_1 = 1$, $\beta_2 = 2$, $N_1 = 3$, $N_2 = 2$, and $\beta = 10$.

(Assignment 9.7#397)

Vector gaussian channel: Consider the vector Gaussian noise channel

$$Y = X + Z,$$

where

$$X = (X_1, X_2, X_3), \quad Z = (Z_1, Z_2, Z_3), \quad Y = (Y_1, Y_2, Y_3),$$
$$\mathbb{E}\|X\|^2 \leq P,$$

and

$$Z \sim \mathcal{N}\left(0, \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 1 & 2 \end{bmatrix}\right).$$

Find the capacity. The answer may be surprising.

(Assignment 9.8#398)

A train pulls out of the station at constant velocity. The received signal energy thus falls off with time as $1/i^2$. The total received signal at time i is

$$Y_i = \frac{1}{i} X_i + Z_i,$$

where $Z_1, Z_2, \ldots$ are i.i.d. $\sim \mathcal{N}(0, N)$. The transmitter constraint for block length n is

$$\frac{1}{n}\sum_{i=1}^{n} x_i^2(w) \leq P, \quad w \in \{1, 2, \ldots, 2^{nR}\}.$$

Using Fano's inequality, show that the capacity C is equal to zero for this channel.

(Assignement-9.9#399)

Additive noise channel: Consider the channel $Y = X + Z$, where X is the transmitted signal with power constraint P, Z is independent additive noise, and Y is the received signal. Let

$$Z = \begin{cases} 0 & \text{with probability } \frac{1}{10}, \\ Z^* & \text{with probability } \frac{9}{10}, \end{cases}$$

where $Z^* \sim \mathcal{N}(0, N)$. Thus Z has a mixture distribution which is a mixture of a gaussian distribution and a degenerate distribution with mass 1 at 0.

1. What is the capacity of this channel? This should be a pleasant surprise.

2. How would you signal to achieve capacity?

(Assignement-9.10#400)

Discrete input, continuous output channel: Let $\Pr\{X = 1\} = p$, $\Pr\{X = 0\} = 1 - p$, and let $Y = X + Z$, where Z is uniform over the interval $[0, a]$, $a > 1$, and Z is independent of X.

(a) Calculate

$$I(X; Y) = H(X) - H(X|Y).$$

(b) Now calculate $I(X; Y)$ the other way by

$$I(X; Y) = h(Y) - h(Y|X).$$

(c) Calculate the capacity of this channel by maximizing over p.

(Assignment-10.1#401)

Consider the case considered in the lecture – finite maximum distortion and finite alphabet. Deduce from the properties of $R(D)$ that it is strictly decreasing until it hits zero.

(Assignment-10.2#402)

Prove that the function

$$S(D) := \min_{P_{Y|X}: E[d(X,Y)] \leq D} I(X;Y)$$

is non-increasing and convex in D.

(Assignment-10.3#403)

One-bit quantization of a single Gaussian random variable: Let $X \sim \mathcal{N}(0, \sigma^2)$ and let the distortion measure be squared error. Here we do not allow block descriptions. Show that the optimum reproduction points for 1-bit quantization are $\pm\sqrt{\frac{2}{\pi}}\sigma$ and that the expected distortion for 1-bit quantization is $\frac{\pi-2}{\pi}\sigma^2$. Compare this with the distortion rate bound $D = \sigma^2 2^{-2R}$ for $R = 1$.

(Assignment-10.4#404)

Lloyd-Max algorithm: Let X be a random variable with probability density function f_X. Suppose we wish to quantise X to L levels. The goal is to minimise the mean squared distortion $E[(X - \hat{X})^2]$. Let $\mu_1^0, \mu_2^0, \cdots, \mu_L^0$ be the L initial representative points. Consider the following iterative procedure. For $k = 1, 2, \ldots$:

- Identify the subsets $A_1^k, A_2^k, \cdots, A_L^k$ that should map to the representatives $\mu_1^{k-1}, \mu_2^{k-1}, \cdots, \mu_L^{k-1}$, respectively.
- Identify the new representatives $\mu_1^k, \mu_2^k, \cdots, \mu_L^k$ for the subsets $A_1^k, A_2^k, \cdots, A_L^k$, respectively.

Solve the two individual steps of the iterative procedure.

(Assignment-10.5#405)

For $x, y \in [0,1]$, show that $(1-xy)^m \leq 1 - y + e^{-mx}$.

(Assignment-10.6#406)

Rate distortion function with infinite distortion.

Find the function $S(D)$ for $X \sim \text{Ber}\left(\frac{1}{2}\right)$ and distortion

$$d(x,\hat{x}) = \begin{cases} 0, & x = \hat{x} \\ 1, & x = 1, \hat{x} = 0 \\ \infty, & x = 0, \hat{x} = 1. \end{cases}$$

(Assignment-10.7#407)

Rate distortion for binary source with asymmetric distortion.

Fix $p(\hat{x}|x)$ and evaluate $I(X;\hat{X})$ and D for $X \sim \text{Ber}\left(\frac{1}{2}\right), d(x,\hat{x}) = \begin{bmatrix} 0 & a \\ b & 0 \end{bmatrix}$

(Assignment-10.8#408)

Rate distortion for uniform source with Hamming distortion: Consider a source X uniformly distributed on the set $\{1, 2, \ldots, m\}$. Find the rate distortion function for the source with Hamming distortion, i.e., $d(x,\hat{x}) = \mathbf{1}\{x \neq \hat{x}\}$.

(Assignment-10.9#409)

Erasure distortion. Consider binary source but with $\hat{X} \in \{1, 0, e\}$ and we use erasure distortion function

$$d(x, \hat{x}) = \begin{cases} 0, & x = \hat{x} \\ 1, & \hat{x} = e \\ \infty, & \text{otherwise.} \end{cases}$$

(Assignment-10.10#410)

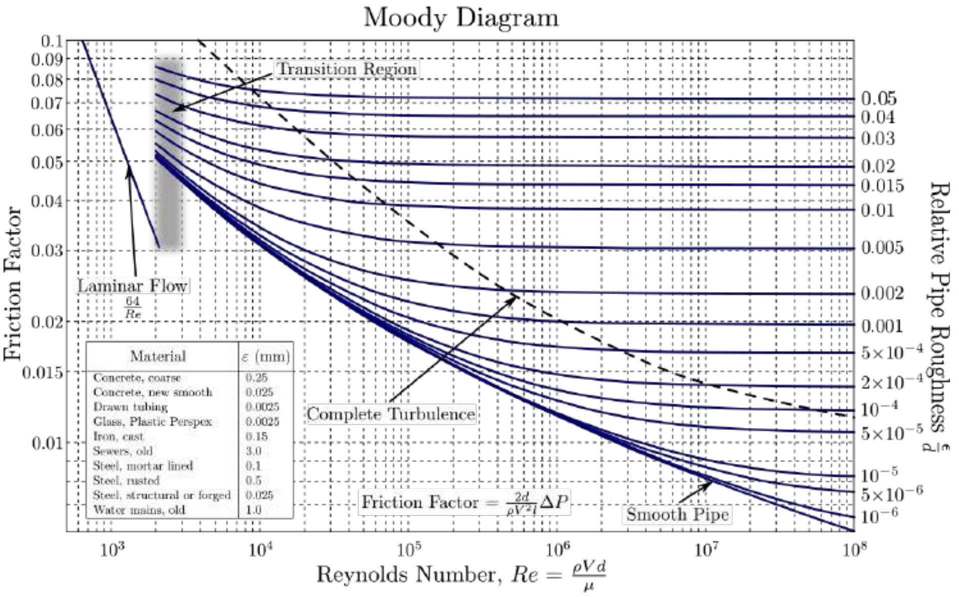
Variational inequality. Verify for positive random variables X that

$$\log E_P(X) = \sup_Q [E_Q(\log X) - D(Q\|P)].$$

This variational characterisation is of fundamental importance in statistical mechanics.

(20250103#1)

Why does the estimate for friction factor give many order of magnitude error for a supposed laminar flow inside a circular pipe in experiments?



Material	ε (mm)
Concrete, coarse	0.25
Concrete, new smooth	0.025
Drawn tubing	0.0025
Glass, Plastic Perspex	0.0025
Iron, cast	0.15
Sewers, old	3.0
Steel, mortar lined	0.1
Steel, rusted	0.5
Steel, structural or forged	0.025
Water mains, old	1.0

The moody chart tells us the variation of friction factor with Reynolds number. The many order of magnitude difference in friction factor hints at the flow becoming turbulent rather than laminar inside the pipe. The flow is inherently unsteady and it cannot be assumed to be 1D for practical scenarios. In long pipes, even small disturbances (e.g., pipe roughness, bends, pumps) trigger turbulence. Laminar flow is unstable at high Re, leading to vortex stretching and energy cascade (Kolmogorov theory). Typically the flow will be laminar for $Re < 2300$, transitional for $2300 < Re < 4000$ and turbulent for $Re > 4000$.

(20250103#2)

What are few of the features of a turbulent flow?

- Irregular
- Large diffusivity: Effect of viscosity is to reduce the gradients; shear stress causes a transport of momentum and it is mediated by viscosity → in molecular diffusion.

  In turbulent flow, the fluid moves chaotically, but this chaos doesn't set the gradient of the average velocity—it's already there in the mean flow. The turbulence fluctuations work to reduce that gradient by mixing things up, and it does so much faster than in non-turbulent flow.

  Molecular diffusion → conductive transport due to viscous terms

  Turbulent diffusion → due to convective terms, not due to viscous terms.

(20250103#3)

How does strain rate vary with length scales in a turbulent flow? How does it relate to the turbulent energy dissipation?

Strain rate in smallest scales must be larger than in larg scales. The strain rate (velocity gradient) increases at smaller scales:

$$S(l) \sim \frac{u(l)}{l}$$

Kolmogorov Scaling for Velocity Fluctuations

$$u(l) \sim (\epsilon l)^{1/3}$$

Resulting strain rate scaling would be

$$S(l) \sim \epsilon^{1/3} l^{-2/3}$$

This shows that strain rate increases as l decreases. Therefore, energy dissipation in smaller scales is much larger. This small scale energy dissipation balances production of energy in large scale flows.

(20250106#4)

Give features of turbulent flows in relation to dissipation, vortical fluctuations and reynolds number

---

- Large dissipation - Kinetic energy dissipated; converted to internal energy.

  Viscous action → converts KE to interal energy. In turbulence → much more pronounced → Why? creates lots of surfaces where velocity gradients are large → amount of dissipation much larger.

- 3D vorticity fluctuationos: Regions of "turbulent" flow need not be turbulent at all. Jet flow - far away - negligible vorticity - non-turbulent flow.

  Intensification of vorticity due to stretching and bending of vortex lines can happen only in $3D$ → turbulent flows are inherently $3D$.

- large Reynolds number: macroscopic Re. Lower Re, perturbations die out due to viscosity. Higher Re, perturbations can grow → flow can become turbulent.

  Reynolds number based on streamwise extent from the leading edge for flow over a flat plate → can help find out the transition location.

(20250106#5)

Can we have a turbulent fluid?

No, we can only have turbulent flows and not turbulent fluids. Turbulence is not a property of the fluid. Likewise inviscid flow and not inviscid fluid. The flow being inviscid/incompressible is a modeling assumption, but turbulence isn't. The flow being turbulent is not a modeling assumption.

(20250106#6)

Why can't we use Navier Stokes equations for rarified flows?

Continuum modeling is correct for turbulent flows. Navier-Stokes is not the right kind of equation to model rarified flows. In rarefied flows, the gas density is low, and the mean free path becomes comparable to or larger than the system scale, violating the continuum assumption. The Boltzmann equation is more appropriate for such conditions.

(20250106#7)

Why does Navier-Stokes equation require continuum assumption?

The Navier-Stokes equations require the continuum assumption because they describe fluid behavior using macroscopic properties like velocity, pressure, and density, which are averaged over a volume. This assumes the fluid is continuous and molecular collisions are frequent enough to define properties like viscosity and thermal conductivity. Without this, the equations cannot accurately represent the system.

(20250106#8)

State the Navier stokes equations in an incompressible flow scenario:

Continuity equation

$$\nabla \cdot \mathbf{v} = 0$$

Momentum equation

$$\rho \left(\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v} \right) = -\nabla p + \mu \nabla^2 \mathbf{v} + \mathbf{f}$$

(20250106#9)

What variables do we seek to obtain through the solution of Navier Stokes equations?

$\boldsymbol{u}(\boldsymbol{x},t)$, $\boldsymbol{p}(\boldsymbol{x},t)$ given initial conditions (IC) and boundary conditions (BC).

(20250106#10)

When will a turbulent flow be a stationary flow?

A turbulent flow is statistically stationary when its statistical properties (e.g., mean velocity, pressure, Reynolds stresses) remain constant over time, even though instantaneous fluctuations persist.

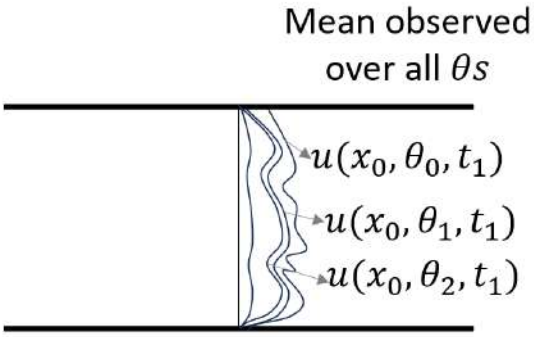
$$\frac{\partial\langle\phi(\mathbf{x},t)\rangle}{\partial t}=0$$

where the averaging used here is ensemble averaging. The ensemble average for a stationary flow will be

$$\frac{1}{N}\sum_{i=1}^{N}u_i(x_0,t_0)=\text{const}$$

for a stationary turbulent flow,

$$\frac{\partial\overline{u_i}}{\partial t}=0,\quad\frac{\partial\overline{p}}{\partial t}=0,\quad\frac{\partial\overline{u_i'u_j'}}{\partial t}=0$$



(20250106#11)

Give the formula for ensemble average and time average:

Ensemble average

$$\langle u \rangle_N = \frac{1}{N}\sum_{i=1}^{N} u_i$$

Time average

$$\langle u \rangle_T = \frac{1}{T}\int_{t_0}^{t_0+T} u(t)dt$$

(20250106#12)

State the Ergodic hypothesis in the context of a turbulent flow:

The Ergodic Hypothesis for turbulent flows states that time averages of flow quantities (e.g., velocity, pressure) over a sufficiently long period converge to their ensemble averages. Mathematically:

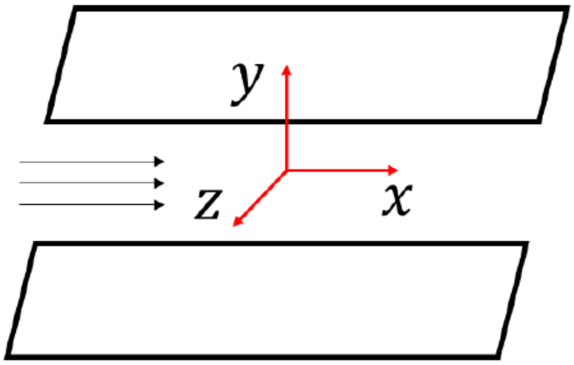
$$\langle \phi(\mathbf{x}) \rangle_{\text{ensemble}} = \lim_{T\to\infty} \frac{1}{T}\int_{t_0}^{t_0+T} \phi(\mathbf{x}, t)\, dt$$

This integration is over a single realization.

Key Assumption: The turbulent flow must be statistically stationary for ergodicity to hold.

(20250106#13)

When is it possible to average flow quantities along a coordinate direction?

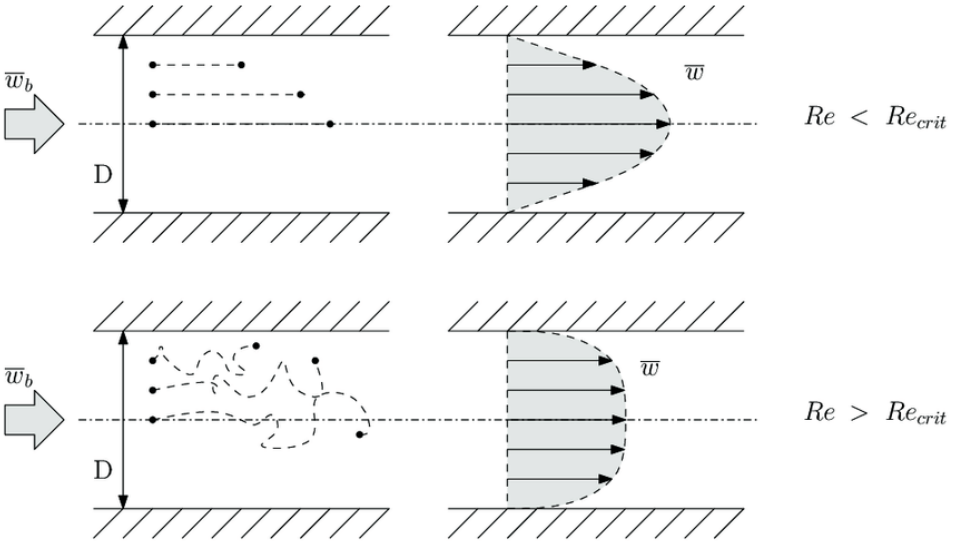


Let's assume a channel flow situation. In its simulation, we can average over the spanwise direction if we assume spanwise periodicity and the flow is homogeneous in that direction.

Similarly if periodic along streamwise direction, we can average along that direction as well. Then in that case, we get $u(y) \to$ field as a function of distance between the plates (averaged over time, and the two periodic directions).

(20250108#14)

Draw laminar and turbulent velocity profile in a pipe. Give a simple equation for the velocity profile in turbulent flow case:



The mean velocity profile for turbulent flow near a wall in a pipe can be approximated by the **1/7th Power Law**:

$$\frac{u(r)}{U_{\max}} = \left(1 - \frac{r}{R}\right)^{1/7}$$

where:

- $u(r)$ is the velocity at radial distance r from the centerline,
- $U_{\max}$ is the maximum velocity (at $r = 0$),
- R is the pipe radius.

Away from the wall, approximate the profile to be U_{max}.

(20250108#15)

What is the overview of steps required to obtain mean velocity fields in a turbulent flow?

- Start with the momentum equation (Newton's 2nd law) for fluid motion.
- Split variables into mean (U) and fluctuating (u') components (Reynolds decomposition).
- Average the equations to model turbulent stresses (τ^R).
- For Newtonian fluids, stress (τ) relates linearly to strain rate (viscosity model) → modeling assumption
- For non-Newtonian/arbitrary fluids, stress modeling is complex (requires closure assumptions).

(20250108#16)

Apply Reynolds decomposition to velocity and pressure fields:

$\boldsymbol{u}(\boldsymbol{x},t) = \boldsymbol{U}(\boldsymbol{x},t) + \boldsymbol{u'}(\boldsymbol{x},t) \quad \text{and} \quad \boldsymbol{p}(\boldsymbol{x},t) = \boldsymbol{P}(\boldsymbol{x},t) + \boldsymbol{p'}(\boldsymbol{x},t)$

Note: Here $\boldsymbol{U}(\boldsymbol{x})$ denotes the time average; If ensemble averaged and the flow is stationary, again $\boldsymbol{U}(\boldsymbol{x})$ is valid as a time average (no time dependence coming from Ergodic hypothesis).

(20250108#17)

Substitute Reynold's decomposition terms to Navier Stokes equation and linearize to get Reynolds equations.

Reynolds decomposition:

$$\boldsymbol{u}(\boldsymbol{x},t) = \boldsymbol{U}(\boldsymbol{x},t) + \boldsymbol{u'}(\boldsymbol{x},t),$$
$$p(\boldsymbol{x},t) = P(\boldsymbol{x},t) + p'(\boldsymbol{x},t).$$

Step 1: Start with incompressible Navier-Stokes

$$\nabla \cdot \boldsymbol{u} = 0,$$
$$\rho\left(\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u}\cdot\nabla\boldsymbol{u}\right) = -\nabla p + \mu\nabla^2\boldsymbol{u}.$$

Step 2: Substitute decomposed variables

$$\nabla \cdot (\boldsymbol{U} + \boldsymbol{u'}) = 0 \quad \Rightarrow \quad \nabla \cdot \boldsymbol{U} = 0,$$
$$\rho\left(\frac{\partial(\boldsymbol{U}+\boldsymbol{u'})}{\partial t} + (\boldsymbol{U}+\boldsymbol{u'})\cdot\nabla(\boldsymbol{U}+\boldsymbol{u'})\right) = -\nabla(P+p') + \mu\nabla^2(\boldsymbol{U}+\boldsymbol{u'}).$$

Step 3: Take time average (Reynolds averaging)

$$\rho\left(\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{U}\cdot\nabla\boldsymbol{U} + \langle\boldsymbol{u'}\cdot\nabla\boldsymbol{u'}\rangle\right) = -\nabla P + \mu\nabla^2\boldsymbol{U}.$$

Step 4: Simplify using averaging rules

$$\langle\boldsymbol{u'}\rangle = 0, \quad \langle\nabla\cdot\boldsymbol{u'}\rangle = 0,$$
$$\langle\boldsymbol{u'}\cdot\nabla\boldsymbol{u'}\rangle = \nabla\cdot\langle\boldsymbol{u'}\otimes\boldsymbol{u'}\rangle.$$

Step 5: Final Reynolds-averaged momentum equation

$$\rho\left(\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{U}\cdot\nabla\boldsymbol{U}\right) = -\nabla P + \mu\nabla^2\boldsymbol{U} - \nabla\cdot(\rho\langle\boldsymbol{u'}\otimes\boldsymbol{u'}\rangle).$$

Step 6: Define Reynolds stress tensor

$$\tau_{\text{turb}} = -\rho\langle\boldsymbol{u'}\otimes\boldsymbol{u'}\rangle,$$

yielding:

$$\rho\left(\frac{\partial \boldsymbol{U}}{\partial t} + \boldsymbol{U}\cdot\nabla\boldsymbol{U}\right) = -\nabla P + \mu\nabla^2\boldsymbol{U} + \nabla\cdot\tau_{\text{turb}}.$$

(20250108#18)

Instead of vector notation, use indicial notation to do the derivation of Reynolds equations for mean flow.

Reynolds decomposition

$$\begin{aligned} u_i(\boldsymbol{x},t) &= U_i(\boldsymbol{x},t) + u_i'(\boldsymbol{x},t),\\ p(\boldsymbol{x},t) &= P(\boldsymbol{x},t) + p'(\boldsymbol{x},t). \end{aligned}$$

Step 1: Incompressible Navier-Stokes in indicial notation

$$\begin{aligned} \frac{\partial u_i}{\partial x_i} &= 0,\\ \rho\left(\frac{\partial u_i}{\partial t} + u_j\frac{\partial u_i}{\partial x_j}\right) &= -\frac{\partial p}{\partial x_i} + \mu\frac{\partial^2 u_i}{\partial x_j \partial x_j}. \end{aligned}$$

Step 2: Substitute decomposed variables

$$\begin{aligned} \frac{\partial(U_i + u_i')}{\partial x_i} &= 0 \quad \Rightarrow \quad \frac{\partial U_i}{\partial x_i} = 0,\\ \rho\left(\frac{\partial(U_i + u_i')}{\partial t} + (U_j + u_j')\frac{\partial(U_i + u_i')}{\partial x_j}\right) &= -\frac{\partial(P + p')}{\partial x_i} + \mu\frac{\partial^2(U_i + u_i')}{\partial x_j \partial x_j}. \end{aligned}$$

Step 3: Expand and take time average

$$\rho\left(\frac{\partial U_i}{\partial t} + U_j\frac{\partial U_i}{\partial x_j} + \left\langle u_j'\frac{\partial u_i'}{\partial x_j}\right\rangle\right) = -\frac{\partial P}{\partial x_i} + \mu\frac{\partial^2 U_i}{\partial x_j \partial x_j}.$$

Step 4: Simplify using averaging rules

$$\begin{aligned} \langle u_i'\rangle = 0, \quad & \left\langle\frac{\partial u_i'}{\partial x_i}\right\rangle = 0,\\ \left\langle u_j'\frac{\partial u_i'}{\partial x_j}\right\rangle &= \frac{\partial}{\partial x_j}\langle u_i' u_j'\rangle. \end{aligned}$$

Step 5: Final Reynolds-averaged momentum equation

$$\rho\left(\frac{\partial U_i}{\partial t} + U_j\frac{\partial U_i}{\partial x_j}\right) = -\frac{\partial P}{\partial x_i} + \mu\frac{\partial^2 U_i}{\partial x_j \partial x_j} - \rho\frac{\partial}{\partial x_j}\langle u_i' u_j'\rangle.$$

Step 6: Define Reynolds stress tensor

$$\tau_{ij}^{\text{turb}} = -\rho\langle u_i' u_j'\rangle,$$

yielding:

$$\rho\left(\frac{\partial U_i}{\partial t} + U_j\frac{\partial U_i}{\partial x_j}\right) = -\frac{\partial P}{\partial x_i} + \mu\frac{\partial^2 U_i}{\partial x_j \partial x_j} + \frac{\partial \tau_{ij}^{\text{turb}}}{\partial x_j}.$$

(20250108#19)

Simplify the Reynolds equation for mean flow for a stationary flow:

For stationary flows,

$$\rho \frac{\partial U_i}{\partial t} = 0$$

Stationary Reynolds-averaged momentum equation:

$$\rho U_j \frac{\partial U_i}{\partial x_j} = -\frac{\partial P}{\partial x_i} + \mu \frac{\partial^2 U_i}{\partial x_j \partial x_j} + \frac{\partial \tau_{ij}^{\text{turb}}}{\partial x_j},$$

Continuity equation (steady)

$$\frac{\partial U_i}{\partial x_i} = 0,$$

where Reynolds stress is defined as

$$\tau_{ij}^{\text{turb}} = -\rho \langle u_i' u_j' \rangle.$$

(20250108#20)

In the Reynold's stress tensor, what are the diagonal and off-diagonal terms called?

Diagonal elements are called Reyno'ds normal stress components while the off-diagonal terms are called Reynold's shear stress components.

(20250108#21)

How many equations and unknowns are there in RANS equations? What causes a mismatch in the two numbers? Does a similar mismatch happen in linear case?

U_i, u_i and P are the unknowns → 7 unknowns.

1 continuity and 3 momentum equations → 4 equations.

This mismatch has occured as a result of the non-linearity of the momentum equations. Simliar problem doesn't happen in linear asen.

Example: Linear equation

$$\nabla^2 f + \alpha f = 0$$

Take fourier transform:

$$(-i\kappa)^2 \hat{f} + \alpha \hat{f} = 0$$

introduces no new terms and hence can be solved with no issues.

(20250108#22)

Why can Reynold's stress tensor be thought of as a collection of line elements in matrix form?

(20250108#23)

When does the components of Reynold's stress tensor become non-zero?

Off-diagonal terms are non-zero when there exists correlation between streamwise and transverse components.

If truly turbulent, then no correlation in any two different directions. Only diagonal terms would exist. Correlation exists because of existence of identifiable turbulent structures, like worm-like, tube-like coherent structures. These coherent structures makes turbulent flow an organized motion rather than random. It looks random because of extreme sensitivity to external input (disturbance). Reynolds stress is treated as a forcing term → forces the structure of mean velocity field.

(20250110#24)

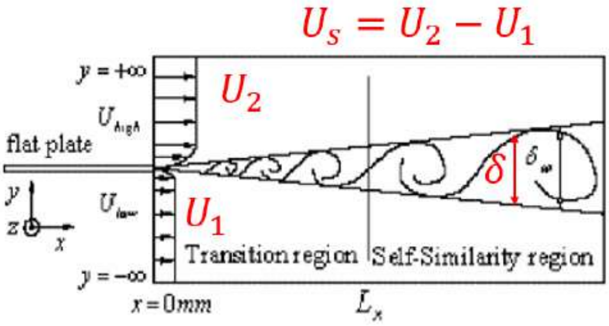
Give examples for canonical flows:

- Free shear flows
  - Mixing layer (shear layer) - plane mixing layer, annular mixing layer
  - Jets - plane jets, round jets
  - Flow past a bluff body - wake: plane wake (circular cylinder), round wake (circular sphere)
- Bounded flows

- wall normal flow
- turbulent channel flow
- turbulent pipe flow

(20250110#25)

Explain briefly about plane mixing layers:



Shear velocity U_s used as the velocity scale for this problem.

Length scale:

$$\delta = \delta(x) \rightarrow 90\% \text{ thickness}$$
$$y_{0.1} \text{ where } U = U_1 + 0.1U_s$$
$$y_{0.9} \text{ where } U = U_1 + 0.9U_s$$

$$\delta(x) = y_{0.9} - y_{0.1}$$

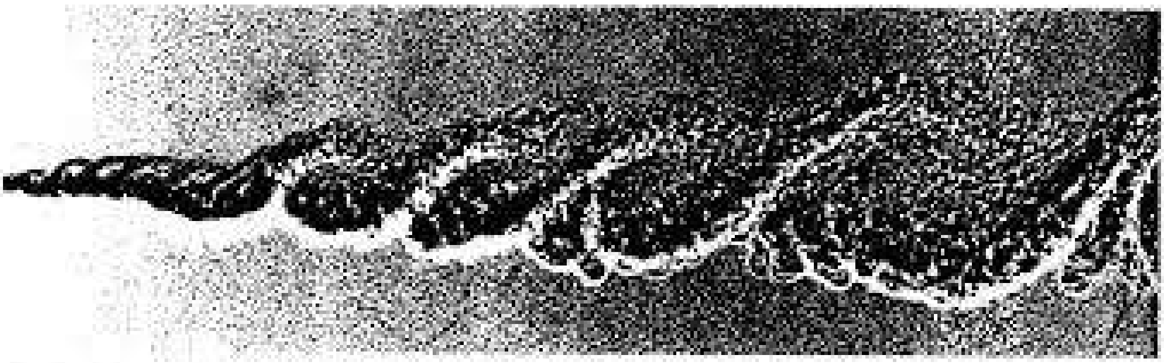


Fig. 5.51. A visualization of the flow of a plane mixing layer. A spark shadow graph of a mixing layer between helium (upper) $U_h = 10.1$ m s^{-1} and nitrogen (lower) $U_l = 3.8$ m s^{-1} at a pressure of 8 atm. (From Brown and Roshko (1974).)

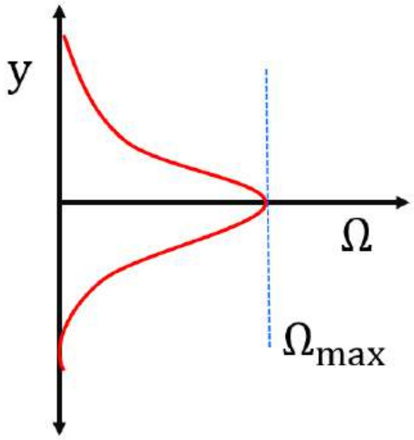
From the figures, we can observe the turbulent structures risen out of Kelvin-Helmholtz instability, indicating that the turbulence can indeed give rise to orderely behavior. But this organization is present in large scales → coherent structures. But it is still complex, has turbulent fluctuations and $3D$ vorticity still exists. Structure emerged is only in largest scales. Small scales eddies need not have this repeating pattern behavior. The coherence is marked by non-zero correlation internally within these flow structures.

(20250110#26)

What is vorticity thickness, say in the context of plane mixing layers?

For a plane mixing layer between two streams with velocities U_1 and U_2 $(U_2 > U_1)$:

- Vorticity: $\Omega(y) = \frac{\mathrm{d}U}{\mathrm{d}y}$
- Maximum vorticity: $\Omega_{\max} = \frac{U_s}{\delta(x)}$
- Velocity scale: $U_s = U_2 - U_1$ (note: U_s is positive)
- Vorticity thickness: $\delta_\omega(x) \equiv \frac{|U_s|}{\Omega_{\max}} = \delta(x)$



The characteristic profiles for a developed mixing layer:

$$U(y) = \frac{U_1 + U_2}{2} + \frac{U_1 - U_2}{2} \tanh\left(\frac{y}{2\delta_\omega}\right) \tag{6}$$

$$\Omega(y) = \frac{U_1 - U_2}{4\delta_\omega} \mathrm{sech}^2\left(\frac{y}{2\delta_\omega}\right) \tag{7}$$

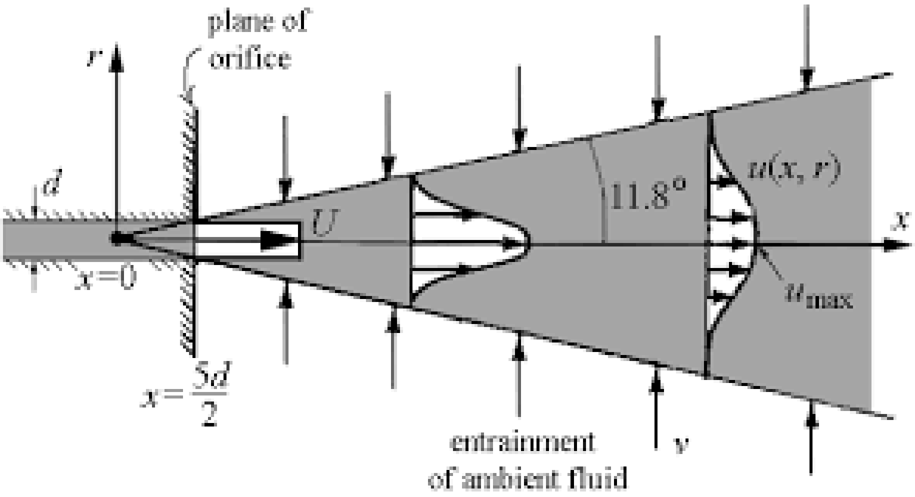
Physical Interpretation

- δ_ω represents the region of concentrated vorticity

- The mixing layer grows linearly downstream: $\delta(x) \sim x$
- For $y = \pm\delta_\omega$, $\Omega \approx \Omega_{\max}/2$
- The negative U_s indicates vorticity of opposite sign to velocity gradient

(20250110#27)
Explain briefly about the scales used in jets and wakes:

Jet Characteristics



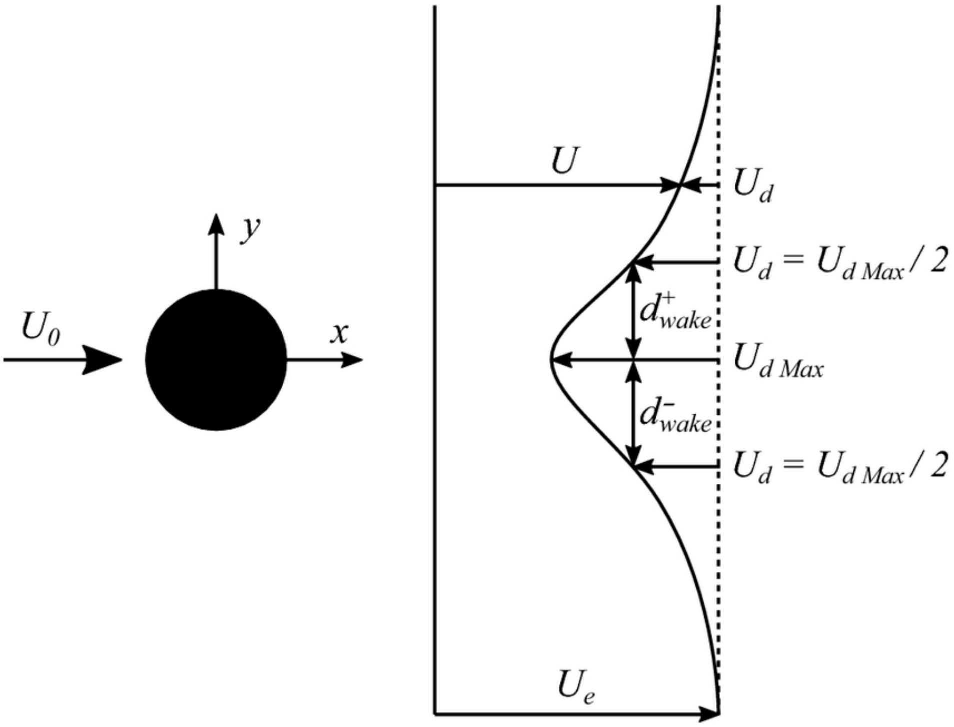
For a turbulent jet with centerline velocity $U_c(x)$:

- **Velocity scale**: U_s, where $U_s = U_c$ if no co-flow and $U_s = U_c - U_a$ if co-flow U_a present.
- **Length scale**: half-width $y_{1/2}$ for plane jets or half-radius $r_{1/2}$ for round jets. The reason why we use half-width is because it is easier to measure half-width than actually measure in the periphery of a jet.

Wake Characteristics

For a wake behind a body of diameter D:

- **Velocity deficit scale**: $\Delta U(o) = U_\infty - U_c(x)$
- **Length scale**: Wake width $\delta(x)$



Wake types: Plane wake, circular/axisymmetric wake

(20250110#28)
Apply scaling to continuity equation for jet/wake system:

Instantaneous Continuity Equation

For incompressible flow:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0$$

Use scaling of derivative terms,

$$\frac{\partial U}{\partial x} \sim O\left(\frac{U_s}{L}\right)$$

where L: streamwise length scale

$$\frac{\partial U}{\partial y} \sim O\left(\frac{U_s}{\delta}\right)$$

Observation $\delta/L << 1$ → thin shear layers Applying the previous two expressions into the continuity equation, we get

$$\frac{\partial v}{\partial y} \sim O\left(\frac{v}{\delta}\right)$$

and hence

$$v \sim O\left(\frac{U_s \delta}{L}\right)$$

Transverse velocity small → in the same vein as thickness of thin shear layer is small.

Note: The perturbations $\langle u'^2 \rangle$, $\langle v'^2 \rangle$, $\langle u'v' \rangle \sim O\left(u_{rms}^2\right)$ (same order fluctuations)

(20250113#29)

Obtain the the time-averaged cross stream momentum equation for jets, wakes and mixing layers:

Cross-Stream Momentum Equation

$$U\frac{\partial V}{\partial x} + V\frac{\partial V}{\partial y} + \frac{\partial \langle u'v' \rangle}{\partial x} + \frac{\partial \langle v'^2 \rangle}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial y} + \nu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2}\right) \tag{8}$$

This is the Reynolds-averaged momentum equation in the wall-normal direction, including contributions from mean convection, Reynolds stresses, pressure gradient, and viscous diffusion.

- Mean convection terms:

  $$U\frac{\partial V}{\partial x} \sim \tilde{U} \cdot \frac{U_s \delta}{L^2}$$

  where

  $$\tilde{U} = \begin{cases} U_s, & \text{jets, mixing layers} \\ U_0, & \text{wakes} \end{cases}$$

- $$V\frac{\partial V}{\partial y} \sim \left(\frac{U_s \delta}{L}\right)^2 \cdot \frac{1}{\delta} = \frac{U_s^2 \delta}{L^2}$$

- Turbulent fluctuation terms (Reynolds stresses):

  $$\frac{\partial \langle u'v' \rangle}{\partial x} \sim \frac{u_{\text{rms}}^2}{L}, \qquad \frac{\partial \langle v'^2 \rangle}{\partial y} \sim \frac{u_{\text{rms}}^2}{\delta} = u_{\text{rms}}^2 \cdot \left(\frac{L}{\delta}\right) \cdot \frac{1}{L}$$

Relative Magnitudes of Convection Terms

- For jets and mixing layers: Mean convection terms scale as

  $$\frac{U_s^2 \delta}{L^2}$$

  and are of the same order.

- For wakes: The streamwise velocity U_0 eventually dominates,

  $$U\frac{\partial V}{\partial x} \sim \frac{U_s U_0 \delta}{L^2}$$

  Initially, $U_0 \sim U_s$, but further downstream $U_0 \gg U_s$.

Viscous Terms

- Streamwise diffusion:

  $$\nu\frac{\partial^2 V}{\partial x^2} \sim \nu \cdot \frac{U_s \delta}{L^3}$$

- Wall-normal diffusion:
$$\nu \frac{\partial^2 V}{\partial y^2} \sim \nu \cdot \frac{U_s \delta}{L \delta^2} = \nu \cdot \frac{U_s}{L \delta}$$
  Hence,
$$\mathcal{O}\left(\nu \frac{\partial^2 V}{\partial y^2}\right) > \mathcal{O}\left(\nu \frac{\partial^2 V}{\partial x^2}\right)$$
  indicating wall-normal viscous effects dominate.

- Writing $\nu U_s \delta / L^3$ in non-dimensional form:
$$\nu \cdot \frac{U_s \delta}{L^3} = \frac{\nu}{U_s L} \cdot \frac{U_s^2 \delta}{L^2} = \frac{1}{Re} \cdot \frac{U_s^2 \delta}{L^2}$$
  In the limit $Re \to \infty$, viscous terms become negligible.

We assume the Reynolds number grows faster than $\delta/L^2 \to 0$, so viscous effects can be dropped from the model.

Comparison of Convection and Fluctuation Terms

- Mean convection term:
$$\frac{U_s^2 \delta^2}{L^2 \delta} = U_s^2 \left(\frac{\delta}{L}\right)^2 \cdot \frac{1}{\delta}$$

- Fluctuation term:
$$\frac{u_{\text{rms}}^2}{\delta} \quad \text{or} \quad \left(\frac{\delta}{L}\right)^2 \cdot \frac{u_{\text{rms}}^2}{U_s^2}$$
  By choosing $u_{\text{rms}} \gg U_s \left(\frac{\delta}{L}\right)$, fluctuations can dominate the mean convection terms. In some relaminarizing scenarios, u_{rms} may decrease significantly, reversing this balance.

Integration of the Cross-Stream Equation

Integrating the cross-stream momentum equation across the shear layer:

$$p + \rho \langle v'^2 \rangle = P_0$$

where P_0 is the pressure outside the turbulent region. The term $\langle v'^2 \rangle$ behaves like a turbulent normal stress, contributing to the total effective pressure. This relation is a form of the turbulent mechanical equilibrium in the transverse direction.

(20250113#30)

Obtain Reynold's averaged streamwise/axial momentum equations for jets, wakes and mixing layers:

$$U\frac{\partial U}{\partial x} + V\frac{\partial U}{\partial y} + \frac{\partial}{\partial x}\langle u'^2\rangle + \frac{\partial}{\partial y}\langle u'v'\rangle = -\frac{1}{\rho}\frac{\partial P}{\partial x} + \nu\left(\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2}\right) \tag{9}$$

This is the Reynolds-averaged streamwise momentum equation, where:

- U, V: mean velocities in the streamwise x and transverse y directions,
- u', v': velocity fluctuations,
- $\langle u'^2\rangle, \langle v'^2\rangle$: normal Reynolds stresses,
- $\langle u'v'\rangle$: shear Reynolds stress,
- ν: kinematic viscosity.

Using Cross-Stream Momentum Balance

From cross-stream momentum analysis, we obtained the approximate pressure relation:

$$P + \rho\langle v'^2\rangle = P_0$$

Differentiating:

$$\frac{\partial P}{\partial x} = -\rho\frac{\partial}{\partial x}\langle v'^2\rangle$$

Substituting this into the streamwise momentum equation yields:

$$U\frac{\partial U}{\partial x} + V\frac{\partial U}{\partial y} + \frac{\partial}{\partial x}\left(\langle u'^2\rangle - \langle v'^2\rangle\right) + \frac{\partial}{\partial y}\langle u'v'\rangle = \nu\left(\frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2}\right)$$

Scaling and Physical Interpretation

- Mean advection: $U\frac{\partial U}{\partial x} \sim U_s^2/L$
- Reynolds stress gradients:

  $$\frac{\partial}{\partial x}\langle u'^2\rangle \sim \frac{u_{\text{rms}}^2}{L}, \quad \frac{\partial}{\partial y}\langle u'v'\rangle \sim \frac{u_{\text{rms}}^2}{\delta}$$

  These are both retained if

  $$\frac{u_{\text{rms}}}{U_s} \sim \mathcal{O}\left(\sqrt{\frac{\delta}{L}}\right)$$

- For jets and mixing layers, often viscous terms are neglected:

  $$U\frac{\partial U}{\partial x} + V\frac{\partial U}{\partial y} + \frac{\partial}{\partial y}\langle u'v'\rangle = 0$$

  This shows that the mean flow is shaped primarily by turbulent fluctuations, especially the Reynolds shear stress $\langle u'v'\rangle$.

- In contrast, for laminar flows the mean velocity field is shaped by viscous diffusion.

Exercises

1. Derive expressions for jets, wakes, and mixing layers using the streamwise momentum equation under different scaling assumptions.
2. Verify whether the approximations made in the cross-stream momentum equation are consistent when substituting $u_{\text{rms}} \sim \mathcal{O}((\delta/x)^{1/2})$.
3. Perform a similar analysis for axisymmetric flows:
   - Round jets (axial and radial components),
   - Round wakes.

Self-Preserving Flows

In turbulent shear flows such as jets and wakes, the notion of self-preservation refers to the statistical similarity of the flow at different streamwise locations.

- The turbulence evolves in such a way that the shape of the velocity and stress profiles remains similar downstream.
- There is no intrinsic streamwise length scale, so similarity variables are used (e.g., $\eta = y/\sqrt{x}$).
- This results in similarity solutions for velocity profiles.

Similarity Solutions

- For a planar jet:

$$U(x,y) = U_s(x) \cdot f\left(\frac{y}{\delta_{1/2}(x)}\right)$$

- For a wake:

$$U_0 - U(x,y) = U_s(x) \cdot f\left(\frac{y}{\delta_{1/2}(x)}\right)$$

  where U_0 is the freestream velocity and U_s is a local velocity scale.

- More generally, we can write:

$$U(x,y) = f\left(\frac{y}{\delta}, \frac{\delta}{L}, \frac{U_s\delta}{\nu}, \frac{U_s}{U_0}\right)$$

  – For jets and mixing layers:

$$\frac{\delta}{L} \ll 1, \quad \frac{U_s\delta}{\nu} = Re \gg 1$$

  These imply that δ/L and $U_s\delta/\nu$ can be neglected.

  – For wakes, the term U_s/U_0 may be retained to distinguish different regions (near, intermediate, far wake).

Summary of Key Profiles

- Streamwise velocity:

$$U(x,y) = U_s \cdot f_1\left(\frac{y}{\delta}\right)$$

- Transverse velocity:

$$V(x, y) = \frac{U_s \delta}{L} \cdot f_2\left(\frac{y}{\delta}\right)$$

- Reynolds shear stress:

$$\langle u'v' \rangle = U_s^2 \cdot g\left(\frac{y}{\delta}\right)$$

These functional forms support the concept of similarity and allow for reduction of PDEs to ODEs using similarity variables in self-preserving flows.

(#31)

- **Absence of a streamwise length scale** often leads to *self-similarity* in the mean velocity profiles.
- This is particularly relevant in free shear flows (e.g., planar jets or wakes) where there is no externally imposed characteristic length along the streamwise direction.
- In such flows, the only relevant length scale is the transverse or cross-stream scale, which itself evolves downstream.

Transverse Length Scale as an Emergent Quantity

- In contrast to the absence of streamwise scales, the transverse length scale (such as the boundary layer thickness $\delta(x)$) *emerges from the solution.*
- For instance, in the classical Blasius boundary layer over a flat plate, the boundary layer thickness $\delta(x) \sim \sqrt{\nu x / U_\infty}$ arises naturally from the similarity analysis of the boundary layer equations.
- This scale is intrinsic to the flow — it is not prescribed externally but rather dictated by the balance of advection and diffusion.

Local Profiles in Self-Similar Flows

- At any given downstream location, the instantaneous velocity field in a turbulent flow may appear highly complex due to the presence of small-scale fluctuations.
- However, when averaged over time (or ensemble), the *mean velocity profiles* exhibit a **self-similar structure** — i.e., they collapse to the same curve when plotted using appropriately scaled coordinates.
- Mathematically, this can be expressed as:

$$U(x, y) = U_s(x) \cdot f\left(\frac{y}{\delta(x)}\right)$$

  where:
  - $U_s(x)$: a local velocity scale,
  - $\delta(x)$: the evolving transverse length scale,
  - $f(\cdot)$: a universal similarity profile.

Implications for Turbulent Flows

- Even in turbulence, where the instantaneous field is chaotic and highly variable, the mean quantities (velocity, Reynolds stress, etc.) can exhibit self-similar behavior.
- This makes self-similarity a powerful tool for:
  - deriving reduced forms of the governing equations,
  - collapsing experimental or simulation data,
  - predicting far-field behavior of turbulent shear flows.

i

(20250115#32)

In general, the mean velocity field $U(x, y)$ in shear flows may be expressed in the form:

$$U(x, y) = f\left(\frac{\delta}{L}, \frac{y}{\delta}, \dots\right)$$

where $\delta(x)$ is a characteristic transverse length scale (e.g., wake width, jet width), and L is a streamwise reference length. In similarity solutions, the streamwise and transverse coordinates collapse into a single similarity variable.

A common factorized form is:

$$U(x, y) = U(x) \cdot F\left(\frac{y}{\delta^*(x)}\right)$$

where $\delta^*(x)$ could be a momentum thickness or other representative width. Similar forms exist for the cross-stream velocity $V(x, y)$, and for Reynolds stresses:

$$\langle u'v' \rangle = U_s^2 \cdot g\left(\frac{y}{\delta}\right)$$

Plane Wake Case

Let us now consider the classical plane wake behind a bluff body. Define the similarity variable:

$$\eta = \frac{y}{\delta(x)}$$

The derivative of η with respect to x is:

$$\frac{\partial \eta}{\partial x} = -\frac{y}{\delta^2}\frac{d\delta}{dx} = -\frac{\eta}{\delta}\frac{d\delta}{dx}$$

Assume a similarity form for the mean streamwise velocity:

$$U(x, y) = U_0 - U_s(x) \cdot f(\eta)$$

where U_0 is the freestream velocity, $U_s(x)$ is a local velocity scale (defect velocity), and $f(\eta)$ is a universal similarity profile.

Compute the derivative:

$$\begin{aligned}\frac{\partial U}{\partial x} &= -\frac{dU_s}{dx} \cdot f(\eta) - U_s(x) \cdot \frac{df}{d\eta} \cdot \frac{\partial \eta}{\partial x} \\ &= -\frac{dU_s}{dx} \cdot f(\eta) + U_s(x) \cdot \frac{df}{d\eta} \cdot \frac{\eta}{\delta} \cdot \frac{d\delta}{dx}\end{aligned}$$

For the Reynolds stress gradient:

$$\frac{\partial}{\partial y}\langle u'v'\rangle = \frac{d}{dy}\left(U_s^2 \cdot g\left(\frac{y}{\delta}\right)\right) = \frac{U_s^2}{\delta} \cdot \frac{dg}{d\eta} \cdot \frac{d\eta}{dy} = \frac{U_s^2}{\delta^2} \cdot g'(\eta)$$

Streamwise Momentum Equation

Ignoring viscous and cross-stream convection terms (justified in far-wake approximation), the momentum equation simplifies to:

$$(U_0 - U_s f)\left[-\frac{dU_s}{dx} f + \frac{U_s}{\delta} \cdot \frac{d\delta}{dx} \cdot \eta f'\right] - \frac{U_s^2}{\delta} g'(\eta) = 0$$

Far-Wake Assumption

In the far wake, $U_s(x) \ll U_0$. As a result, higher-order terms in U_s/U_0 can be neglected. Moreover, the similarity solution depends only on $\eta = y/\delta$, and not separately on x and y. To ensure this, coefficients of the ODE must be independent of x, leading to scaling constraints.

Define:

$$U_s(x) = Ax^n, \quad \delta(x) = Bx^m$$

Substitute into the expressions to identify conditions for x-independence.
Constraint 1:

$$-\frac{U_0\delta}{U_s^2} \cdot \frac{dU_s}{dx} \sim \text{const} \Rightarrow -\frac{U_0 Bx^m}{A^2x^{2n}} \cdot (Anx^{n-1}) = -n\frac{U_0 B}{A}x^{m-n-1} \Rightarrow m = n+1$$

Constraint 2:

$$\frac{U_0}{U_s} \cdot \frac{d\delta}{dx} \sim \text{const} \Rightarrow \frac{U_0}{Ax^n} \cdot mBx^{m-1} = m\frac{U_0 B}{A}x^{m-n-1} \Rightarrow m = n+1$$

Thus, both constraints are satisfied if:

$$\delta(x) \sim x^{n+1}, \quad U_s(x) \sim x^n$$

Neglecting Cross-Stream Convection

In the plane wake, the term $V\partial U/\partial y \sim \mathcal{O}(U_s^2/\delta)$ is small compared to $U\partial U/\partial x \sim \mathcal{O}(U_0U_s/x)$ under far-wake assumptions.

Additional Constraint: Momentum Deficit Conservation

The total drag or momentum deficit in the wake must be conserved downstream. This global constraint typically leads to an additional integral condition:

$$\int_{-\infty}^{\infty} [U_0 - U(x,y)]\, dy = \text{const} \Rightarrow U_s(x) \cdot \delta(x) = \text{const} \Rightarrow Ax^n \cdot Bx^{n+1} = \text{const} \Rightarrow n + m = 0$$

Combining this with $m = n+1 \Rightarrow n+n+1 = 0 \Rightarrow n = -\frac{1}{2}, m = \frac{1}{2}$

Therefore,

$$U_s(x) \sim x^{-1/2}, \quad \delta(x) \sim x^{1/2}$$

These are the classical similarity scalings for a plane turbulent wake.

(20250115#33)

The Reynolds-averaged x-momentum equation for incompressible turbulent flow is given by:

$$U\frac{\partial U}{\partial x} + V\frac{\partial U}{\partial y} + \frac{\partial}{\partial y}\langle u'v'\rangle = 0$$

where U, V are mean velocities, and $\langle u'v'\rangle$ is the Reynolds shear stress.

Assuming U_0 is a constant freestream velocity, we rewrite the equation in terms of the velocity defect $u_d = U - U_0$:

$$U\frac{\partial(U-U_0)}{\partial x} + V\frac{\partial(U-U_0)}{\partial y} + \frac{\partial}{\partial y}\langle u'v'\rangle = 0 \tag{1}$$

Now consider the identity:

$$(U-U_0)\left(\frac{\partial U}{\partial x} + \frac{\partial V}{\partial y}\right) = 0$$

This follows from continuity and the fact that $U = U_0$ outside the wake, where there is no flow divergence.

Rewriting Derivatives

We expand total derivatives of composite products:

$$\frac{\partial}{\partial x}\left[U(U-U_0)\right] = U\frac{\partial(U-U_0)}{\partial x} + (U-U_0)\frac{\partial U}{\partial x}$$
$$\frac{\partial}{\partial y}\left[V(U-U_0)\right] = V\frac{\partial(U-U_0)}{\partial y} + (U-U_0)\frac{\partial V}{\partial y}$$

Combining:

$$\frac{\partial}{\partial x}\left[U(U-U_0)\right] + \frac{\partial}{\partial y}\left[V(U-U_0)\right] = U\frac{\partial(U-U_0)}{\partial x} + V\frac{\partial(U-U_0)}{\partial y} \tag{2}$$

Substituting equation (2) into equation (1), we get:

$$\frac{\partial}{\partial x}\left[U(U-U_0)\right] + \frac{\partial}{\partial y}\left[V(U-U_0)\right] + \frac{\partial}{\partial y}\langle u'v'\rangle = 0$$

Integral Form in y-Direction

Integrating across the entire transverse direction:

$$\int_{-\infty}^{\infty} \frac{\partial}{\partial x} [U(U - U_0)] \, dy + \int_{-\infty}^{\infty} \frac{\partial}{\partial y} [V(U - U_0)] \, dy + \int_{-\infty}^{\infty} \frac{\partial}{\partial y} \langle u'v' \rangle dy = 0$$

Using the divergence theorem and boundary conditions:

$$\frac{d}{dx} \int_{-\infty}^{\infty} U(U - U_0) dy + V(U - U_0)|_{-\infty}^{+\infty} + \langle u'v' \rangle|_{-\infty}^{+\infty} = 0$$

Outside the turbulent region (in the freestream), we assume:

- $\langle u'v' \rangle = 0$
- $U = U_0$
- $V = 0$

Thus, the last two boundary terms vanish, yielding:

$$\frac{d}{dx} \int_{-\infty}^{\infty} U(U - U_0) dy = 0 \Rightarrow \int_{-\infty}^{\infty} U(U - U_0) dy = \text{constant}$$

This conserved quantity reflects the streamwise momentum deficit in the wake and is used for determining far-field scaling.

Applicability to Jets and Mixing Layers

- This relation *does not apply* to mixing layers, since:

$$\lim_{y \to -\infty} U = U_1, \quad \lim_{y \to +\infty} U = U_2 \Rightarrow U \not\to U_0$$

- For wakes: $U \to U_0$ on both sides $\Rightarrow$ integral condition applies.
- For jets: $U \to 0$ as $y \to \pm\infty$, and the conserved quantity is:

$$\frac{d}{dx} \int_{-\infty}^{\infty} U^2 dy = 0$$

Far-Wake Scaling

Assume similarity form:

$$U(x, y) = U_0 - U_s(x) f(\eta), \quad \eta = \frac{y}{\delta(x)}$$

Compute the momentum deficit:

$$\int_{-\infty}^{\infty}[U_0 - U(x,y)] \cdot U(x,y)dy = \int_{-\infty}^{\infty} U_s f(\eta)[U_0 - U_s f(\eta)] \cdot \delta \, d\eta$$

In the far wake, $U_s \ll U_0$, so we approximate:

$$\int_{-\infty}^{\infty} U_0 U_s f(\eta) \cdot \delta \, d\eta = \text{const} \Rightarrow U_0 U_s \delta \int_{-\infty}^{\infty} f(\eta) d\eta = C_3$$

Assume:

$$U_s(x) = Ax^m, \quad \delta(x) = Bx^n \Rightarrow U_0 Ax^m \cdot Bx^n = \text{const} \Rightarrow m + n = 0$$

From earlier similarity analysis, we also have:

$$m = n + 1 \Rightarrow n + 1 + n = 0 \Rightarrow 2n = -1 \Rightarrow n = -\frac{1}{2}, \quad m = \frac{1}{2}$$

Conclusion:

$$U_s(x) \sim x^{-1/2}, \quad \delta(x) \sim x^{1/2}$$

These are the classical self-similar scalings for a plane turbulent wake.

(20250115#34)

Scaling Behavior of Free Shear Flows

We characterize several canonical free shear flows by their streamwise decay of centerline velocity $U_s(x)$ and growth of shear layer thickness $\delta(x)$. The similarity solutions suggest power-law behavior of the form:

$$U_s(x) \sim x^n, \quad \delta(x) \sim x^m$$

A summary of these exponents for various flows is provided below:

Flow Type	n (Scaling of U_s)	m (Scaling of δ)	$n+m$
Plane wake	$-\frac{1}{2}$	$\frac{1}{2}$	0
Round wake	$-\frac{2}{3}$	$\frac{1}{3}$	$-\frac{1}{3}$
Mixing layer	0	1	1
Plane jet	$-\frac{1}{2}$	1	$\frac{1}{2}$
Round jet	-1	1	0

Implication for Reynolds Number Based on Shear Layer Thickness

Define a local Reynolds number based on the similarity velocity scale and shear layer thickness:

$$\mathrm{Re}_\delta = \frac{U_s(x) \cdot \delta(x)}{\nu} \sim x^{n+m}$$

- **If** $n + m > 0$, then $\mathrm{Re}_\delta \to \infty$ downstream — turbulence strengthens.
- **If** $n + m = 0$, then Re_δ is constant — sustained turbulence.
- **If** $n + m < 0$, then $\mathrm{Re}_\delta \to 0$ — turbulence decays, potential for *relaminarization.*

Relaminarization in Round Wakes

In the case of the round wake:

$$n = -\frac{2}{3}, \quad m = \frac{1}{3} \Rightarrow n + m = -\frac{1}{3} \Rightarrow \mathrm{Re}_\delta \sim x^{-1/3}$$

Thus, Re_δ decreases with streamwise distance. This suggests a tendency for the round wake to undergo *relaminarization*, especially far downstream, due to decreasing turbulent intensity. This is a fundamental difference compared to plane wakes or jets, where turbulence is typically sustained or grows.

Remark on Round Jets

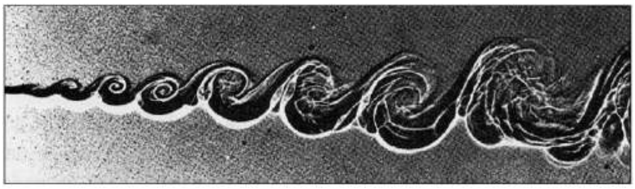
For a round jet:

$$n = -1, \quad m = 1 \Rightarrow n + m = 0 \Rightarrow \mathrm{Re}_\delta = \text{constant}$$

The similarity assumption in the round jet is known to be valid not only in the far field but also relatively close to the nozzle, due to the strong axisymmetric nature of the spreading. The constancy of Re_δ in this case supports a statistically steady turbulent structure.

(20250117#35)

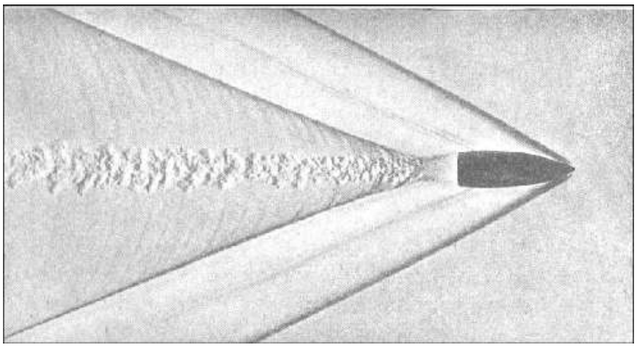
Explain this image from Brown and Roshko (1974):



- **Inner Layer of a Shear Layer is Turbulent:** In free shear flows such as mixing layers, the region near the center of the shear layer—often referred to as the inner layer—is where the turbulence is most intense. Brown and Roshko observed that even in this chaotic region, there exists a degree of structure.
- **Presence of a Characteristic Wavelength:** Despite the inherently random nature of turbulence, the flow exhibits dominant, repeating structures that possess a particular spatial wavelength. These are associated with large-scale coherent vortices that appear periodically in the shear layer.
- **Implications for Controlled Mixing:** The identification of a dominant wavelength means that the mixing layer is not entirely random. Instead, the organized vortices can be manipulated or enhanced using external perturbations at specific frequencies, leading to controlled mixing—important for engineering applications like combustion and jet noise reduction.
- **Turbulence is not Purely Chaotic:** A major insight from Brown and Roshko's work is that turbulence includes elements of order. Large-scale coherent structures dominate the momentum and energy transport, especially in the initial development region of shear layers. These structures follow patterns, and are not merely the outcome of random fluctuations.
- **Perturbation Growth and Nonlinear Transition:** Small perturbations in the flow may initially grow linearly. However, as they increase in amplitude, nonlinear effects become significant, and the disturbances start rolling up into large vortical structures. This vortex roll-up marks a transition to nonlinear dynamics and the onset of organized turbulence.
- **Role of the Wall (in context):** Though Brown and Roshko primarily studied free shear layers, the presence of walls can modify the behavior of these coherent structures significantly. Walls can introduce additional shear, alter the development of vortices, or suppress/enhance specific modes. Understanding this is critical in wall-bounded turbulence studies.
- **Interaction with Turbulent Flow via Coherent Structures:** If the turbulent flow contains coherent structures with identifiable wavelengths, then it becomes possible to externally excite or suppress these structures using appropriately timed inputs (e.g., acoustic forcing, mechanical actuators). This concept underlies the control strategies in flow control and turbulence management.

(20250117#36)

Explain the flow in the wake of a speeding bullet:



- **Lack of Large-Scale Organization:** Unlike the turbulent mixing layer discussed by Brown and Roshko, the flow in the wake of a high-speed projectile (e.g., a bullet) does not exhibit the same level of coherent, large-scale vortical structures. The turbulence here appears less organized, possibly due to higher-speed effects, compressibility, or different initial conditions.

- **Possibly Helical Rather than Wavy Structures:** In this regime, instead of quasi-two-dimensional vortex roll-up with a clear periodic wavelength, the instabilities may manifest as helical or spiral modes. These structures can arise in axisymmetric flows and are particularly relevant in the wakes of slender bodies, where azimuthal instabilities dominate over planar shear-layer instabilities.

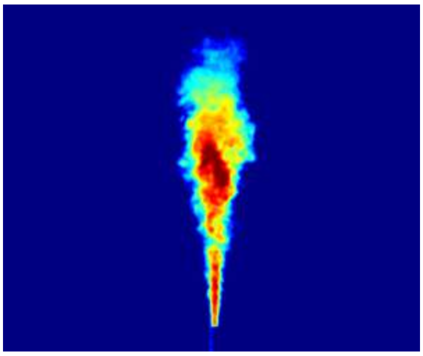
- **Sharp Interface Indicates Strong Boundary Layer Dynamics:** The observation of a very sharp interface between the turbulent and non-turbulent regions suggests that significant processes are occurring within a very thin boundary layer. This implies strong velocity and scalar gradients near the interface, necessitating very fine-scale resolution to accurately capture the dynamics.

- **Importance of Small Scales:** The fine structure near the interface means that small-scale turbulence plays a crucial role in the mixing and transport processes. Unlike flows dominated by large coherent structures, this regime requires careful attention to dissipation, viscosity, and small-scale modeling (e.g., in LES or DNS). Capturing these features is essential for understanding phenomena like shock-boundary layer interactions or energy dissipation in high-speed flows.

(20250117#37)

Describe the nature of turbulence in a round jet:

- **Presence of a Broad Range of Scales:**
  Unlike the previous cases—such as the organized structures in mixing layers or small-scale dominated regions like the boundary of a bullet—the turbulent round jet exhibits



activity across a *continuous spectrum of scales*, from the large-scale jet core to the finest dissipative structures. This indicates a fully developed turbulent flow where the energy cascades through an inertial range from large eddies to small eddies.

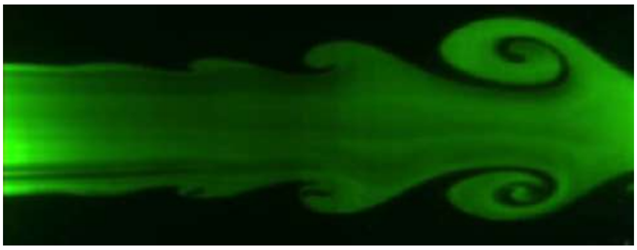
- **Jet Entrainment as a Key Mechanism:**
  One of the most fundamental processes in turbulent jets is *entrainment*—the drawing in of surrounding ambient fluid into the jet due to turbulent mixing. Entrainment governs the spread and decay of the jet and is essential in determining how momentum, mass, and scalar quantities (e.g., temperature or concentration) are transported.

- **Dye Visualization of Entrainment:**
  In PLIF experiments, the jet fluid is typically marked with a fluorescent dye. This allows clear optical visualization of mixing: the dyed fluid originates from the jet, while the undyed fluid comes from the surrounding ambient. Entrainment is then observed by tracking the invasion of dye-marked structures into previously undyed regions.

- **Filamentous Structures at Multiple Scales:**
  The entrainment process does not occur as a smooth blending of fluids but through the formation of *filamentous structures*—thin, thread-like regions of dyed fluid penetrating into the ambient. These filaments occur over a wide range of length scales, from large coherent structures at the jet boundary to thin scalar layers governed by small-scale straining. Their visibility across scales highlights the multiscale nature of turbulent mixing and scalar transport in round jets.

(20250117#38)

Describe the turbulence observed in nozzle exit jets:



- **Formation and Evolution of Toroidal Vortex Rings:**
  At the exit of a nozzle, especially during the startup phase of a pulsed or synthetic jet, toroidal vortex rings are formed due to the roll-up of the shear layer at the interface between the jet and the ambient fluid. These vortex rings grow in size as more circulation is entrained into them from the shear layer.

  - **Instability of Vortex Rings:**
    Once a vortex ring has formed and rolled up, it becomes susceptible to three-dimensional perturbations. The *fastest growing instability mode* in such rings is often the **sinuous mode**, characterized by asymmetric undulations of the ring's core, resembling a snake-like motion around the toroidal axis.

  - **Dependence on Ring Slenderness:**
    The growth rate and nature of the instability depend on the ring's *slenderness ratio*, defined as the ratio of the core thickness (vortex cross-section diameter) to the ring radius. Slender rings tend to be more unstable to sinuous modes.

- **Nonlinear Effects and Vortex Roll-up:**
  As the amplitude of the instability grows beyond the linear regime, nonlinear effects dominate. These lead to the breakdown of the smooth vortex ring into more complex, turbulent structures. This process can be understood as the transition from linear instability to vortex roll-up and breakdown.

- **Possible Link to Widnall Instability:**
  The term "Wigner instability" may refer to the **Widnall instability**, a known instability mechanism for vortex rings and tubes. This refers to the azimuthal instability modes that develop on a vortex filament and can lead to core oscillations and eventual breakdown.

- **Linear Instability:**
  In the early stages, the growth of perturbations can be described by linear stability theory, where disturbances grow exponentially with time:

  $$\text{Perturbation amplitude } \sim e^{\sigma t}$$

  where σ is the growth rate of the most unstable mode.

- **Time Scale of Breakdown:**
  Two time scales are important here:

  - *Convective time scale:* Based on the bulk velocity of the jet and the characteristic nozzle dimension, this gives an impression of abrupt vortex breakdown when observed in the lab frame.

  - *Vorticity-based time scale:* This is based on the local rotation rate (vorticity) in the vortex core. Since the vorticity is high in the vortex ring, this time scale reveals that the instability grows more gradually from the perspective of the vortex dynamics.

- **Large Vorticity in the Core:**
  The vorticity in the vortex ring core is significantly large due to strong shear during formation. This makes the ring highly sensitive to instabilities and facilitates rapid transition to turbulence.

(20250117#39)

Explain the turbulence observed in rocket exhausts, like in this PSLV launch:

- The jet emerging from a PSLV (Polar Satellite Launch Vehicle) nozzle has an extremely high Reynolds number, typically on the order of $Re \sim 10^7$ or higher. This is due to:
  - Large characteristic velocity U (high exhaust speeds from the nozzle),
  - Large length scale L (nozzle diameter),
  - Small kinematic viscosity ν of the exhaust gases.
- At such high Reynolds numbers, one would expect the flow to exhibit extremely fine-scale turbulence and a wide inertial range in the energy cascade. However, the surprising observation is:

> *"The evolution of the PSLV jet at enormously large Reynolds numbers develops very much like a modestly large, low Reynolds number jet."*

- This observation is rooted in the fact that:
  - **The large-scale structure of the jet is dominated by coherent structures and large eddies**, especially in the initial development region. These structures are often similar across a wide range of Reynolds numbers, due to self-similar behavior.
  - **Non-dimensional quantities such as U/U_c and x/D (centerline velocity, axial location normalized by nozzle diameter)** tend to collapse onto similar profiles even across different Reynolds numbers.
  - **In the near field of the jet**, where coherent vortices dominate entrainment and mixing, the Reynolds number has only a secondary effect. The flow topology is governed more by inviscid instabilities (like Kelvin–Helmholtz instability) than by viscosity.
- **Implication:** The near-field jet dynamics—especially entrainment, shear layer growth, and vortex roll-up—can be effectively studied using moderate-Reynolds number laboratory jets, making them excellent models for certain aspects of rocket plume behavior.

- **However:** At larger distances downstream or when investigating fine-scale turbulence or mixing at the molecular level, the extremely high Reynolds number becomes significant. Differences in the dissipation range and turbulence spectrum will become more evident.

- Thus, from a global structural and engineering perspective, the high-Re jet (like PSLV) appears to "develop like" a moderate-Re jet in terms of jet shape, spread rate, and mean velocity profiles in the near field.

(20250117#40)

How does the free shear flow through a nozzle exit differ from a confined flow such as in a pipe?

- **Pipe Flow:**
  - In fully developed pipe flow, the flow exhibits a characteristic velocity profile depending on the flow regime (laminar or turbulent).
  - **Laminar Flow:**
    - The velocity profile is parabolic, described by the Hagen-Poiseuille law:

      $$u(r) = U_{\max}\left(1 - \left(\frac{r}{R}\right)^2\right)$$

      where $U_{\max}$ is the centerline velocity and R is the pipe radius.
    - The velocity varies smoothly from zero at the wall (due to the no-slip condition) to a maximum at the centerline.
    - This smooth variation creates a relatively thick shear layer, extending from the wall to the center.
  - **Turbulent Flow:**
    - The profile becomes flatter in the center, with sharp velocity gradients confined near the wall.
    - The bulk of the fluid moves at nearly the same velocity, often modeled using the power-law or log-law profiles.
    - The shear layer is now thin and concentrated in the near-wall region, often called the *turbulent boundary layer*.

- **Nozzle Flow:**
  - Unlike pipe flow, a nozzle accelerates the flow, often from a large plenum into a smaller exit area, forming a jet.
  - At the nozzle exit, the flow profile is nearly uniform across the core, with a relatively thin shear layer separating the jet core from the surrounding stationary or coflowing fluid.

  - This shear layer is thin regardless of whether the flow is laminar or turbulent.
  - In both laminar and turbulent nozzle flows:
    - The core velocity is approximately constant (plug-like profile).
    - The velocity drops rapidly across a small radial extent—this forms the shear layer.
  - The thin shear layer is the region of strong velocity gradient and is critical for instability development (e.g., Kelvin–Helmholtz instability), which can lead to vortex roll-up and turbulence transition in the jet.

(20250117#41)

Explain the similarities and differences observed in turbulent round jet experiments of Panchapakesan et.al. and Hussein et. al.:

- **Reynolds Number (Re) Considerations:**
  - Several experiments have been conducted to study the characteristics of turbulent round jets at different Reynolds numbers:
    - Panchapakesan & Lumley (1993): $Re = 11000$
    - Hussein, Capp & George (1994): $Re = 95500$
- **Why was $Re = 11000$ chosen by Panchapakesan & Lumley?**
  - At Reynolds numbers near 10000, an important transition in the turbulent mixing process occurs.
  - This transition is often referred to as the *merging transition*, where the initially distinct shear layers from the nozzle exit merge into a single turbulent shear layer.
  - To ensure the study captured the effects post-transition while still maintaining moderate flow control and facility requirements, a Reynolds number slightly above the critical range was chosen: $Re = 11000$.
  - This value allows detailed exploration of turbulence dynamics while avoiding complexities introduced at very high Reynolds numbers.
  - In contrast, Hussein & collaborators selected a much larger $Re = 95500$ to study the asymptotic, high-Reynolds-number behavior of turbulent jets.
- **Measurement Techniques:**
  - **Laser Doppler Anemometry (LDA):** A non-intrusive optical technique used to measure velocity components in the jet. It provides high spatial and temporal resolution.
  - **Hot-Wire Anemometry (HWA):** A moving hot-wire probe is employed to obtain detailed velocity fluctuation statistics. It involves placing a heated wire in the flow and measuring the cooling rate, which relates to local velocity.

(20250117#42)

Detail the results obtained from Panchapakesan et. al. and Hussein et. al.:

- As a turbulent round jet evolves downstream from a nozzle, its velocity field spreads outward due to turbulent mixing and entrainment.
- An important concept in turbulent jet analysis is the idea of **self-preserving** or **self-similar** behavior.

Self-Preserving Flow:

- If a flow is **self-preserving**, the velocity profile, when appropriately normalized, has the same shape at every axial location.
- In a self-similar flow, quantities such as velocity and jet width can be described by scaling laws, and the normalized profiles collapse onto a single curve.
- This implies that the spatial evolution of the jet becomes independent of initial conditions once the flow has transitioned to the self-preserving regime.

Centerline Velocity Decay:

- The centerline velocity, denoted $U_c(x)$, represents the maximum velocity along the axis of the jet.
- In the self-preserving region, the centerline velocity decays inversely with downstream distance x:
$$U_c(x) \sim \frac{1}{x}$$
- A more refined empirical expression often used in experimental fits is:
$$U_c(x) = \frac{B_u}{(x - x_0)/d}$$

  where:

  - B_u is a constant determined from experiments,
  - x_0 is the virtual origin of the jet (accounting for the shift due to initial development),
  - d is the nozzle diameter.

Experimental Parameters for Turbulent Round Jets

Interpretation:

- The slope s describes the rate at which the jet spreads downstream.

Table 1: Experimental Parameters for Turbulent Round Jets

Reference	**Re**	**Slope (s)**	$\boldsymbol{\beta}$
Panchapakesan & Lumley (1993)	11000	0.096	6.06
Hussein, Capp & George (1994) - HW	95500	0.102	5.9
Hussein, Capp & George (1994) - LDA	95500	0.094	5.8

- β is a profile shape parameter or spreading coefficient related to the radial distribution of velocity.
- Despite significant differences in Reynolds numbers, the spreading rates and profile parameters remain within a relatively narrow range, indicating robust self-preserving behavior.

(20250120#43)

Draw the variation of f with $\xi = r/r_{1/2}$ for a round jet. Explain why the plot looks like that.

(20250120#44)

Plot the variation of transverse velocity with $\xi = r/r_{1/2}$. Explain why the plot looks like that.

(20250120#45)

What does $\langle v \rangle / U_0$ as a function of $\xi = r/r_{1/2}$ decreasing and becoming negative indicate?

(20250120#46)

What remains invariant for a turbulent jet?

(20250318#47)

Explain the role of transverse pressure gradient in a turbulent jet.

(20250120#48)

What is required to maintain the momentum flux of a turbulent jet?

(20250120#49)

Explain why momentum flux in a turbulent jet doesn't change with downstream location:

(20250120#50)

Why does entrainment occur in jets?

(20250120#51)

State the result from Spalding's paper which gives an extent as to how much of the ambient flow is getting entrained.

(20250120#52)

Plot the variation of $\langle u_i u_j \rangle / \langle U^2 \rangle$ vs $\xi = r/r_{1/2}$. How does the plot look for $\langle u'^2 \rangle$, $\langle v'^2 \rangle$ and $\langle w'^2 \rangle$?

(20250120#53)

Write down the expressions for momentum flux, mass flux and energy flux. Explain how they vary with downstream distance.

(20250120#54)

What two essential processes are important in the dynamics of wall-bounded flows unlike free shear flow scenarios?

(20250120#55)

Give examples for wall-bounded flows:

(20250120#56)

Sketch a channel flow diagram. State the assumptions involved.

(20250120#57)

What are the two ways in which Re is defined for channel flows in literature and elsewhere?

(20250120#58)

What are the laminar and turbulent Reynolds numbers Re for channel flow case?

(20250120#59)

Give the continuity, x-momentum and y-momentum equations for channel flows.

(20250122#60)

What drives a channel flow?

Channel flows are typically pressure driven flows confined between two parallel plates, typically assumed to be infinitely long in the streamwise direction.

$$\frac{dP}{dx} = 0$$

Pressure decreases linearly along the length of the channel, pushing the flow forward. Pressure gradient acts as a forcing mechanism here.

(20250122#61)

Write the expression for x-momentum equation for a channel flow. Obtain the expression for shear stress from that equation.

(20250122#62)

In the expression for shear stress in a channel flow, explain which terms dominate in the near wall region and in the region far away from the walls:

(20250122#63)

Write the simplified form of expression obtained from Navier-Stokes equation for a channel flow. Obtain $u(z)$ from that expression.

$$\mu\frac{d^u}{dz^2} = \frac{dP}{dx}$$

from which we obtain via integration,

$$u(z) = \frac{1}{2\mu}\frac{dP}{dz}\left(h^2 - z^2\right)$$

where h is the half-height of the channel.

(20250122#64)

Write x momentum equation (streamwise direction) for channel flow:

$$0 = \nu\frac{\partial^2 U}{\partial y^2} - \frac{\partial}{\partial y}\langle u'v'\rangle - \frac{1}{\rho}\frac{\partial P}{\partial x}$$

This is the mean flow equation in the streamwise (x) direction, where

- $\nu\partial^2 U/\partial y^2$ is the viscous diffusion term which denotes the momentum transfer due to molecular viscosity.
- $-\partial\langle u'v'\rangle/\partial y$ denotes Reynolds stress gradient, representing the effect of turbulence. The Reynolds stress here denotes the turbulent momentum transport.
- $-(1/\rho)\partial P/\partial x$ denotes pressure gradient, which is the driving force for the flow.

This equation denotes steady state balance of forces in the x-direction due to the LHS being 0 instead of a time derivative term.

(20250122#65)

How is shear stress related to pressure gradient in this case? Why this relation?

The total shear stress is coming from the contributions due to mean velocity gradient and due to the turbulent shear stress, which can be expressed as

$$\tau(y) = \rho\nu\frac{dU}{dy} - \rho\langle u'v'\rangle$$

Substituting this into

$$0 = \nu\frac{\partial^2 U}{\partial y^2} - \frac{\partial}{\partial y}\langle u'v'\rangle - \frac{1}{\rho}\frac{\partial P}{\partial x}$$

one obtains

$$\frac{d\tau}{dy} = \frac{dP}{dx}$$

The pressure gradient is opposed by the shear stress gradient to maintain equilibrium.

(20250122#66)

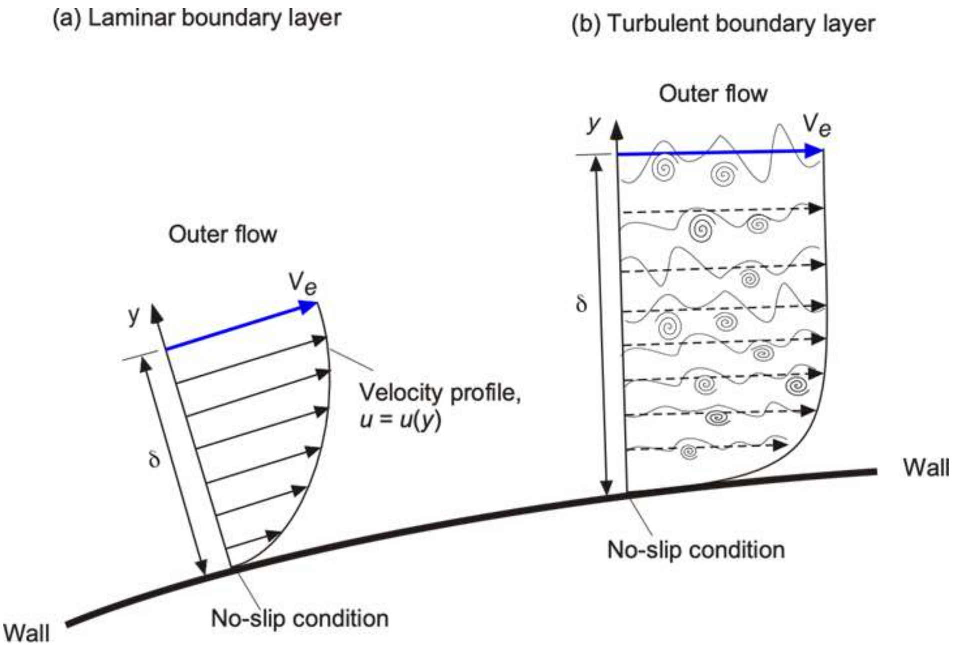
How does the contribution to the total shear stress by viscous shear stress and turbulent shear stress vary in the bulk flow vs near wall region?

Near wall, viscous stresses dominate due to very strong mean flow velocity gradients $dU/dy \neq 0$ close to the wall.

$$\tau \approx \rho\nu\frac{dU}{dy}$$

In the bulk flow, turbulent stress dominates.

$$\tau \approx -\rho\langle u'v'\rangle$$



(20250122#67)

What can be said about the shear stress profile for a turbulent channel flow?

We'll end up with a linear shear stress profile. This can be seen from this relation obtained from turbulent channel flow:

$$\frac{d\tau}{dy} = \frac{dP}{dx}$$

Integrating wrt y, we get

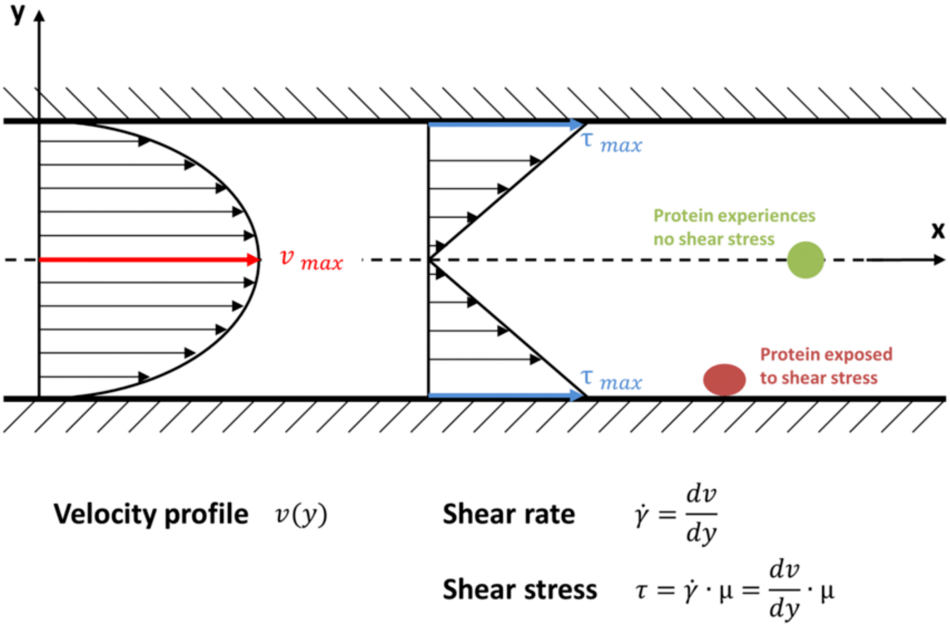
$$\tau(y) = \frac{dP}{dx}y + C$$

where C is a constant.

Using boundary condition at the centerline $y = H/2$, where H is the channel height, symmetry requires $\tau = 0$. Thus,

$$\tau(y) = \frac{dP}{dx}\left(y - \frac{H}{2}\right).$$

The stress varies linearly from τ_w (wall shear stress) at $y = 0$ to 0 at $y = H/2$.


y
x
v_{max}
τ_{max}
τ_{max}
Protein experiences no shear stress
Protein exposed to shear stress
Velocity profile $v(y)$
Shear rate $\dot{\gamma} = \frac{dv}{dy}$
Shear stress $\tau = \dot{\gamma} \cdot \mu = \frac{dv}{dy} \cdot \mu$

(20250122#68)

Why does symmetry at centerline require $\tau = 0$?

(20250122#69)

What happens to turbulent shear stress right at the wall?

Due to no-slip boundary condition, at $y = 0$, we have

$$U = 0$$

and $u' = 0$ will also hold. Thus at the wall, we have $\langle u'v' \rangle = 0$, i.e., Reynolds shear stress (turbulent shear stress) is zero at the wall under no-slip boundary conditions.

(20250122#70)

Explain how turbulent fluctuations act like extra diffusion:

In laminar flow, molecular viscosity μ is the only mechanism that smooths out velocity gradients (diffusion of momentum). This leads to a parabolic velocity profile (e.g., in Poiseuille flow).

In turbulent flow, chaotic fluctuations (u, v) create additional ”mixing” of momentum, similar to an enhanced diffusivity.

This turbulent diffusion flattens the velocity gradient in the bulk flow (far from the wall), making the profile more uniform (not parabolic).

Analogy: Just like viscosity smoothens gradients in laminar flow, turbulent fluctuations do the same—but much more aggressively.

(20250122#71)

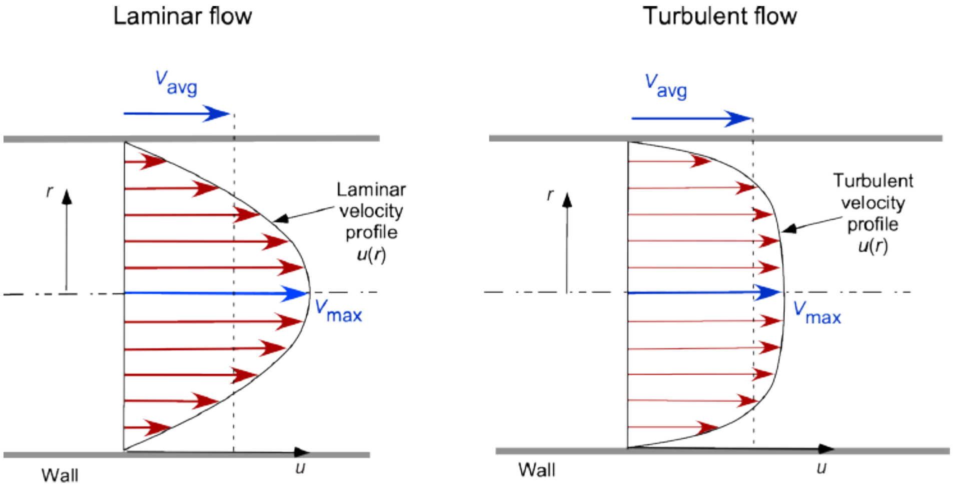
Why is the velocity gradient sharper near the wall for a turbulent channel flow as opposed to a laminar case?

Near the wall, turbulence is suppressed (because viscosity dominates at small scales). However, the no-slip condition ($u = 0$ at the wall) still applies, so the flow must adjust rapidly.

Turbulent fluctuations try to smooth out the velocity gradient (making it less steep).

Wall shear stress resists this smoothing (because the flow must stick to the wall).

Result: The velocity gradient near the wall becomes much steeper than in laminar flow (higher du/dy), meaning much higher shear stress at the wall.



(20250122#72)

Where in the turbulent boundary layer does turbulence and viscosity compete?

Even very close to the wall, turbulent fluctuations are still present (though weaker than in the bulk flow).

This region (the viscous sublayer) is where viscosity and turbulence compete.

Farther from the wall, turbulence dominates, and the velocity profile becomes nearly flat (logarithmic in turbulent boundary layers).

(20250122#73)

Why do we model near-wall and far-wall regions separately?

Near-wall region:

- Viscous effects dominate.
- Velocity gradient is steep.
- Requires fine resolution (in simulations) or special models (e.g., wall functions) because turbulence is damped but still influential.

Far-wall (bulk) region:

- Turbulent diffusion dominates.
- Velocity profile is flatter.
- Can be modeled with simpler turbulence models (e.g., $k - \epsilon$, RANS).

(20250122#74)

Derive the mean velocity profile in turbulent flow near and away from the walls using dimensional arguments (scaling laws) without invoking Navier-Stokes equations.

Consider the near-wall region first. This corresponds to the viscous sublayer, where viscosity dominates.

We assume that very close to the wall, turbulence is suppressed (with viscous forces dominating over turbulent inertia).

Define velocity and length scale based on the key players deciding the dynamics of the flow very close to the wall, mainly wall shear stress and viscosity:

- Friction velocity (u_τ): Defined by the wall shear stress, this comes up by taking $\rho u_\tau^2 = \tau_w$,
$$u_\tau = \sqrt{\frac{\tau_w}{\rho}}$$
- Viscous length scale: how far viscosity affects the flow (δ_ν):
$$\delta_\nu = \frac{\nu}{u_\tau}$$
This relation follows from the definition of a local Reynolds number Re_τ in the vicinity of this viscous sub-layer region, i.e.
$$Re_\tau = \frac{u_\tau \delta}{\nu} = \frac{\delta}{\delta_\nu}$$
This Re_τ is different from $Re = \bar{U}h/\nu$ or $Re = \bar{U}(2\delta)/\nu$ which is based on geometry and imposed conditions. Those are sort of external or outer Re, while Re_τ is an internal Re, which we call as friction Re. The mean velocity in this region can only depend on u_τ, ν and distnace from the wall (y). The only dimensionless combination is:
$$u^+ = \frac{u}{u_\tau} \qquad y^+ = \frac{yu_\tau}{\nu}$$
where we call y^+ as wall units. Since there are no other variables, we get
$$u^+ = y^+ \qquad \text{(Viscous sublayer law)}$$
This gives linear profile very close to the wall Away from the wall, we have the log law region (where turbulence dominates).

We assume that viscosity no longer matters in this region - turbulent mixing dominates here. The velocity gradient now depends on wall shear stress and distance from the wall.

In this case, the velocity scale would still be u_τ. But the length scale would be y instead of δ_ν.

The mean velocity gradient can only depend on u_τ and y. So we have
$$\frac{dU}{dy} \propto \frac{u_\tau}{y}$$

from which we obtain the log law following logarithmic velocity profile, given by

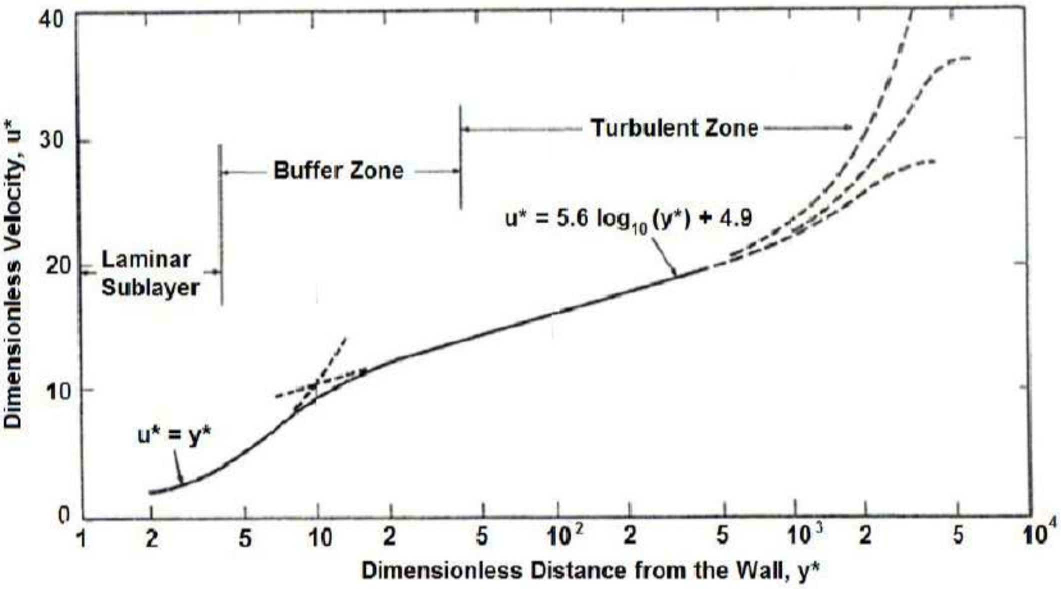
$$u^+ = \frac{1}{\kappa} \ln y^+ + B$$

where B is an empirical constant and κ is the Von Karman constant which is empirically ~ 0.41.

The transition between these regions is empirically determined to be around $y^+ \approx 30$.

Table 2: Turbulent Boundary Layer Structure

Region	**Dominant Physics**	**Velocity Profile**	**Key Scaling**
Viscous Sublayer ($y^+ < 5$)	Viscosity dominates	$u^+ = y^+$	u_*, ν
Buffer Layer ($5 < y^+ < 30$)	Transition zone	(Blending)	(Empirical)
Log-Law Region ($y^+ > 30$)	Turbulence dominates	$u^+ = \frac{1}{\kappa} \ln y^+ + B$	u_*, y



(20250122#75)

Give an expression for $\tau(y)$ in channel flows:

$$\tau(y) = \tau_w \left(1 - \frac{y}{\delta}\right) = \rho\nu \frac{\partial U}{\partial y} - \rho\langle u'v' \rangle$$

(20250122#76)

Plot the contribution of viscous stresses and shear stresses towards total stress:

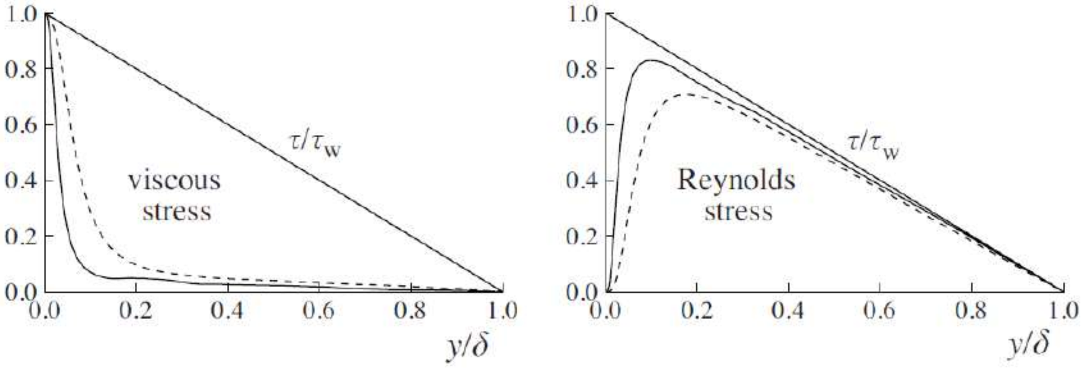


Fig. 7.3. Profiles of the viscous shear stress, and the Reynolds shear stress in turbulent channel flow: DNS data of Kim *et al.* (1987): dashed line, Re = 5,600; solid line, Re = 13,750.

(20250122#77)

What happens to the region where viscous stresses are dominant as Re increases? What is the reason for this behavior?

The viscous sublayer becomes thinner and thinner. This can be explained by looking at the length scale δ_ν.

$$\delta_\nu = \frac{\nu}{u_\tau}$$

where $u_\tau = \sqrt{\tau_w/\rho}$.

At higher Re, mean flow velocity U increases, so inertial forces grow. Wall shear stresses τ_w will increase because of the rapider growth of mean flow velocity from 0 at the wall to larger U away from the wall. This results in the increase of u_τ. Since $\delta_\nu \propto 1/u_\tau$, the viscous sublayer becomes thinner and thinner with Re.

This is one of the reasons why high Re flows can be treated as mostly inviscid, but need viscosity to satisfy no-slip condition.

(20250122#78)

Plot fractional contributions of viscous and shear stresses as a function of wall units.

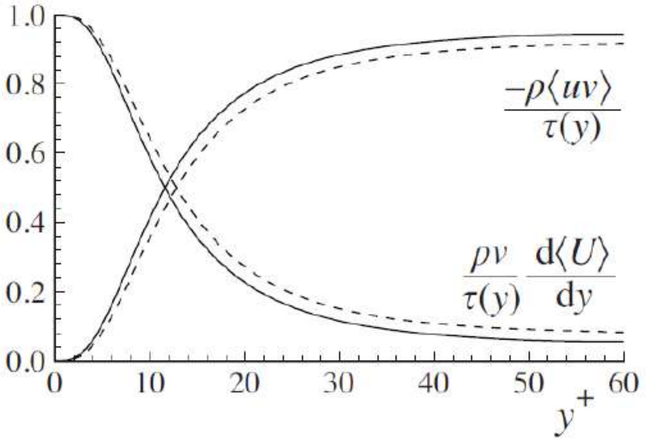


Fig. 7.4. Profiles of the fractional contributions of the viscous and Reynolds stresses to the total stress. DNS data of Kim *et al.* (1987): dashed lines, Re = 5,600; solid lines, Re = 13,750.

(20250122#79)

Explain briefly about the symmetry group theory perspective of deriving turbulent boundary layer velocity profile:

The log-law can also be derived using Lie symmetry groups applied to the Navier-Stokes equations. Here's the gist:

- Assume the mean flow is invariant under certain transformations (e.g., scaling y and U).
- Apply the infinitesimal generator of the symmetry group to the Reynolds-averaged Navier-Stokes (RANS) equations.
- Result: The only self-similar solution satisfying the symmetries is the log-law.

It proves that the log-law is mathematically consistent with Navier-Stokes and that turbulence has universal scaling far from walls. It says that the log-law as a fundamental property of turbulent scaling, not just an empirical fit. It's a rare example of an exact solution for turbulent flows, even though turbulence is chaotic!

(20250122#80)

Obtain a functional form of expression for du^+/dy^+ for the inner region:

Parameters that we already know → $\rho, \nu, \delta, dP/dx$. Because of balance between pressure gradient and shear stresses,

$$u_\tau = \left(\frac{\delta}{\rho}\frac{dP}{dx}\right)^{1/2}$$

This comes from

$$\begin{aligned} &\tau(y) = \tau_w \left|1 - \frac{y}{\delta}\right| \\ &\frac{d\tau}{dy} = \frac{\tau_w}{\delta} \\ &\text{But,} \quad \frac{d\tau}{dy} = \frac{dP}{dx} \\ \implies &\tau_w = \delta\frac{dP}{dx} \\ &\text{We also know,} \ \ \rho u_\tau^2 = \tau_w \\ \implies &u_\tau = \sqrt{\frac{\delta}{\rho}\frac{dP}{dx}} \end{aligned}$$

$$u^+ = \frac{u}{u_\tau} = F\left(\frac{y}{\delta}, Re_\tau\right)$$

Re_τ is how viscosity ν comes into the equation.

Instead of talking about u, alternatively,

$$\frac{1}{u_\tau}\frac{dU}{dy} = \frac{1}{y}\Phi\left(\frac{y}{\delta}, \frac{y}{\delta_\nu}\right)$$

Length scale chosen as distance from the wall y. Choosing δ makes no sense as the region we consider has height $<<\delta$. Choosing δ_ν gives results not matching with the experimental results. Go with y itself.

$$\frac{dU}{dy} = \frac{u_\tau}{y}\Phi\left(\frac{y}{\delta}, \frac{y}{\delta_\nu}\right)$$

For the inner layer region,

$$\Phi_I = \Phi(y/\delta_\nu) = \lim_{y/\delta \to 0} \Phi(y/\delta, y/\delta_\nu)$$

which gives

$$\frac{dU}{dy} = \frac{u_\tau}{y}\Phi_I(y/\delta_\nu)$$

Casting it in terms of wall units,

$$\frac{du^+}{dy^+} = \frac{1}{y^+}\Phi_I(y^+)$$

(20250124#81)

Obtain expressions for mean velocity profiles in a boundary layer and explain the different regions involved:

The mean velocity gradient in wall-bounded turbulent flow can be expressed as:

$$\frac{du}{dy} = \frac{u_\tau}{y}\Phi\left(\frac{y}{\delta}, \frac{y}{\delta_\nu}\right)$$

where:

- $u_\tau = \sqrt{\tau_w/\rho}$ is the friction velocity.
- δ is the outer length scale (e.g., boundary layer thickness).
- $\delta_\nu = \nu/u_\tau$ is the viscous length scale.

Near the Wall: $y^+ \ll 1$

In the near-wall region, define dimensionless variables:

$$u^+ = \frac{U}{u_\tau}, \quad y^+ = \frac{yu_\tau}{\nu}$$

The velocity gradient becomes:

$$\frac{du^+}{dy^+} = \Phi^+(y^+)$$

As $y^+ \to 0$, using Taylor expansion:

$$U^+(y^+) = U^+(0) + y^+ \left.\frac{du^+}{dy^+}\right|_{y^+=0} + \mathcal{O}(y^{+2})$$

With the no-slip condition $U^+(0) = 0$, we get:

$$U^+(y^+) = y^+ \left.\frac{du^+}{dy^+}\right|_{y^+=0}$$

To show $\left.\frac{du^+}{dy^+}\right|_{y^+=0} = 1$:

$$\tau_w = \mu \left.\frac{dU}{dy}\right|_{y=0} = \mu\frac{u_\tau}{\delta_\nu} \left.\frac{du^+}{dy^+}\right|_{y^+=0}$$

$$\Rightarrow \rho u_\tau^2 = \rho\delta_\nu\frac{u_\tau}{\nu} \left.\frac{du^+}{dy^+}\right|_{y^+=0} \Rightarrow \left.\frac{du^+}{dy^+}\right|_{y^+=0} = 1$$

Thus,

$$U^+ = y^+ \quad \text{for } y^+ \ll 1 \quad \text{(Viscous Sublayer)}$$

Logarithmic Layer: $y/\delta \gg 1$

For large y^+, experiments and theory suggest:

$$\Phi^+(y^+) = \frac{1}{\kappa} \Rightarrow \frac{dU^+}{dy^+} = \frac{1}{\kappa y^+}$$

Integrating:

$$U^+ = \frac{1}{\kappa} \ln\bigl(y^+\bigr) + B$$

This logarithmic law holds in the **overlap region**, where both inner and outer scalings are valid.

Outer Layer and Matching

In the outer region:

$$\frac{dU}{dy} = \frac{u_\tau}{\delta} \Phi\left(\frac{y}{\delta}\right)$$

Integrating from y to δ:

$$\frac{U_0 - U}{u_\tau} = \int_{y/\delta}^{1} \Phi(\eta)\, d\eta$$

In the overlap region:

$$\Phi(y/\delta) = \Phi^+(y^+) = \frac{1}{\kappa} \Rightarrow \frac{U - u_\tau}{u_\tau} = \frac{1}{\kappa} \ln\left(\frac{y}{\delta}\right) - A$$

From the inner side:

$$\frac{U}{u_\tau} = \frac{1}{\kappa} \ln\left(\frac{y u_\tau}{\nu}\right) + B = \frac{1}{\kappa} \ln(y) + \frac{1}{\kappa} \ln\left(\frac{u_\tau}{\nu}\right) + B$$

Matching the two:

$$\frac{1}{\kappa} \ln\left(\frac{u_\tau}{\nu}\right) + B = -\frac{1}{\kappa} \ln(\delta) - A + 1 \Rightarrow \frac{1}{\kappa} \ln\left(\frac{u_\tau \delta}{\nu}\right) + B + A = 1$$

Layer Classification

- **Viscous Sublayer:** $0 < y^+ < 5$
- **Buffer Layer:** $5 < y^+ < 30$
- **Log Layer:** $30 < y^+$, $y/\delta < 0.3$
- **Outer Layer:** $y/\delta \approx 1$

Note on CFD Practice In numerical simulations, when the first mesh point from the wall is at $y^+ \approx 10$, it lies in the buffer layer. Hence, the viscous sublayer is not captured unless finer resolution is used.

(20250203#82)

What is the key difference in the mean flow equation of laminar vs turbulent flows?

Turbulent flow is fundamentally different from laminar flow because of the appearance of Reynolds stresses ($-\rho\langle u_i' u_j'\rangle$) in the governing equations. These represent momentum transport by turbulent fluctuations and arise from the non-linear inertial effects in the $u \cdot \nabla u$ term in the Navier-Stokes when decomposed to mean+fluctuations.

Laminar flow Navier-Stokes equation is

$$\rho\left(\frac{\partial \mathbf{U}}{\partial t} + \mathbf{U}\cdot\nabla\mathbf{U}\right) = -\nabla P + \mu\nabla^2\mathbf{U}$$

while for turbulent flows, the mean flow equation (RANS) is

$$\rho\left(\frac{\partial \overline{\mathbf{U}}}{\partial t} + \overline{\mathbf{U}}\cdot\nabla\overline{\mathbf{U}}\right) = -\nabla\overline{P} + \mu\nabla^2\overline{\mathbf{U}} - \rho\nabla\cdot\langle\mathbf{u}'\mathbf{u}'\rangle$$

Table 3: Comparison of Laminar and Turbulent Flow Characteristics

Feature	Laminar Flow	Turbulent Flow
Stresses	Only viscous	Viscous + Reynolds
Mixing	Molecular diffusion	Strong turbulent mixing
Energy Dissipation	Slow	Rapid (via eddy cascade)

(20250203#83)

Give the expression for Reynolds stress in matrix form:

In matrix notation,

$$\boldsymbol{\tau}^{\text{turb}} = -\rho\langle\mathbf{u}'\mathbf{u}'\rangle = -\rho\begin{pmatrix}\langle u'u'\rangle & \langle u'v'\rangle & \langle u'w'\rangle \\ \langle v'u'\rangle & \langle v'v'\rangle & \langle v'w'\rangle \\ \langle w'u'\rangle & \langle w'v'\rangle & \langle w'w'\rangle\end{pmatrix}$$

In index notation,

$$\tau_{ij}^{\text{turb}} = -\rho\langle u_i' u_j' \rangle$$

(20250203#84)

Explain some properties of Reynold's stress tensor

Some key properties of Reynold's stress tensor:

- It is a second-order tensor representing turbulent momentum transport.
- Symmetry $\tau_{ij}^R = \tau_{ji}^R$ → only 6 independent components (3 diagonal and 3 off-diagonal).
- It is a tensor as opposed to a matrix. It obeys coordinate transformation rules, and has physical meaning (turbulent momentum flux), unlike a matrix which is just an array with no inherent physics.
- Coordinate invariance:
  - First invariant (Linear):

    $$\text{tr}\left(\tau^R\right) = \tau_{ii}^R = -\rho\left(\langle u'^2\rangle + \langle v'^2\rangle + \langle w'^2\rangle\right)$$

    Trace → sum of diagonal terms → turbulent kinetic energy (TKE).
  - Second invariant (Quadratic):

    $$\frac{1}{2}\left[\left(\text{tr}\left(\tau^R\right)\right)^2 - \text{tr}\left(\left(\tau^R\right)^2\right)\right]$$

    related to anisotropy intensity.
  - Third invariant (cubic)

    $$\det\left(\tau^R\right)$$

    Determines the shape of turbulence (pancake vs cigar-like eddies).
- Anisotropy
  - Isotropic part:

    $$\frac{1}{3}\text{tr}\left(\tau^R\right)\delta_{ij}$$

    invariant, like pressure. This is the same in all directions → no directional preference.
  - Anisotropic part:

    $$\tau_{ij}^R - \frac{1}{3}\text{tr}\left(\tau^R\right)\delta_{ij}$$

    Changes with orientation → distinguishes turbulence from laminar flow.

    Example: $\langle u'^2\rangle \neq \langle v'^2\rangle$ in shear flows.

(20250203#85)

How is isotropic turbulence different from anisotropic turbulence?

In isotropic turbulence, we have all diagonal terms in Reynolds stress terms being equal, $\langle u'^2\rangle = \langle v'^2\rangle = \langle w'^2\rangle$. All the off-diagonal terms are 0 due to velocity correlations being uncorrelated, $\langle u_i' u_j'\rangle = 0$ for $i \neq j$. They don't have any specific preference for shear direction, which leads to 0 shear correlation. Momentum transport happens equally in all directions. It occurs rarely in reality (requires no mean shear). Anisotropic turbulence dominates near walls or shear layers and causes directional mixing (e.g., $\langle u'v'\rangle \neq 0$ in boundary layers). Turbulence isn't random. Its directional memory (anisotropy) is encoded in τ_{ij}^R.

(20250203#86)

Write expression for Boussinesq approximation

$$\tau_{ij}^{\text{turb}} \approx \frac{2}{3}\rho k \delta_{ij} - \rho \nu_t \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right)$$

where the first term on RHS is the istropic part and the second term is the anisotropic part.

(20250203#87)

What's the expression for turbulence (or turbulent) kinetic energy?

$$k = \frac{1}{2}\left(\langle u'^2\rangle + \langle v'^2\rangle + \langle w'^2\rangle \right)$$

For isotropic turbulence,

$$k = \frac{1}{2}\langle u_i' u_i'\rangle = \frac{3}{2}\langle u'^2\rangle$$

(20250203#88)

Obtain expression for momentum equation as isotropic and deviatoric parts:

Reynold's stress tensor splits into isotropic and deviatoric (anisotropic) parts:

$$\langle u_i' u_j' \rangle = \underbrace{\frac{2}{3} k \delta_{ij}}_{\text{Isotropic}} + \underbrace{a_{ij}}_{\text{Deviatoric}}$$

where the deviatoric part is

$$a_{ij} = \langle u_i' u_j' \rangle - \frac{2}{3} k \delta_{ij}$$

The isotropic part is the energy containing istropic motions, while the deviatoric part is the anisotropic momentum transport. Momentum equation - original form with the Reynolds stresses:

$$\frac{Du_i}{Dt} = \nu \nabla^2 u_i - \frac{1}{\rho} \frac{\partial P}{\partial x_i} - \frac{\partial}{\partial x_j} \langle u_i' u_j' \rangle$$

Substituted form using the decomposition into isotropic and deviatoric part would be

$$\frac{Du_i}{Dt} = \nu \nabla^2 u_i - \frac{1}{\rho} \frac{\partial}{\partial x_i} \left(P + \frac{2}{3} \rho k \right) - \frac{\partial}{\partial x_j} a_{ij}$$

(20250203#89)

What are the implications of the isotrpoic and deviatoric terms in the momentum equation?

The isotropic term acts like a turbulent pressure:

$$\frac{2}{3} k \delta_{ij}$$

The deviatoric term captures the directional anisotropy:

$$a_{ij}$$

The new pressure term is the modified pressure

$$P^* = P + \frac{2}{3} \rho k$$

(20250203#90)

What happens to the momentum equation with modified pressure when the spatial derivative of deviatoric term is 0?

When we include the Reynolds stress decomposition in the momentum equation:

$$\frac{Du_i}{Dt} = \nu\nabla^2 u_i - \frac{1}{\rho}\frac{\partial \tilde{P}}{\partial x_i} - \frac{\partial a_{ij}}{\partial x_j}$$

where $\tilde{P} = P + (2/3)k$ is the modified pressure.

If we set $\partial a_{ij}/\partial x_j = 0$, the equation reduces to

$$\frac{Du_i}{Dt} = \nu\nabla^2 u_i - \frac{1}{\rho}\frac{\partial \tilde{P}}{\partial x_i}$$

This is mathematically identical to the laminar Navier-Stokes equation (with $\tilde{P}$ replacing P. Thus, without it, turbulence would behave like laminar flow with rescaled pressure.

Concept	**Mathematical Expression**	**Physical Meaning**
Modified pressure	$\tilde{P} = P + \frac{2}{3}\rho k$	Turbulence adds "effective pressure"
Anisotropic stress term	$\frac{\partial a_{ij}}{\partial x_j}$	Sole source of turbulent mixing effects
Turbulence intensity	$k = \frac{3}{2}u'^2_{\mathrm{rms}}$	Direct measure of fluctuation energy

(20250203#91)

For homogeneous isotropic turbulence, won't the deviatoric term be 0 anyways? Then isn't the momentum equation like a modmified version of laminar flow?

(20250203#92)

Despite just a contributor to turbulence pressure, what is the significance of k?

- Intensity Measure: k quantifies the amplitude of velocity fluctuations

$$u'_{\mathrm{rms}} = \sqrt{\frac{2}{3}k}$$

- Turbulence Scaling: Low k → Weak turbulence, while high k → strong mixing. In short, it defines the fluctuation amplitude u'_{rms}.
- Reynolds stress scaling: the anisotropic stress scales with k

$$|a_{ij}| \sim k$$

Measuring k (e.g., via PIV/LDA), directly indicates turbulence intensity, independent of how it appears in the equations.

(20250203#93)

Explain with an example as to how one can arrive at a model for Reynolds stress:

For a turbulent flow with velocity $\boldsymbol{u} = \boldsymbol{U} + \boldsymbol{u}'$ (mean $\boldsymbol{U}$ + fluctuations $\boldsymbol{u}'$), instantaneous kinetic energy would be

$$E(\mathbf{r}, t) = \frac{1}{2}(\mathbf{u} \cdot \mathbf{u}) = \frac{1}{2}u_i u_i$$

while averaged kinetic energy would be

$$\langle E \rangle = \underbrace{\frac{1}{2}(\mathbf{U} \cdot \mathbf{U})}_{\overline{E}} + \underbrace{\frac{1}{2}\langle \mathbf{u}' \cdot \mathbf{u}' \rangle}_{k}$$

where $\overline{E}$ is the kinetic energy of the mean flow and k is the turbulence kinetic energy (TKE) quantifying fluctuation intensity.

To model the Reynolds stress tensor:

$$\tau_{ij}^{\text{turb}} = -\rho \langle u_i' u_j' \rangle$$

we need two key ingredients, the amplitude scale and the shape/anisotropy. We use k to come up with amplitude scale,

$$|\tau_{ij}^{\text{turb}}| \sim \rho k$$

while for shape/anisotropy, we use a dimensionless tensor b_{ij},

$$b_{ij} = \frac{\langle u_i' u_j' \rangle}{2k} - \frac{1}{3}\delta_{ij}$$

which follows the trace condition $b_{ii} = 0$, which ensures consistency with k. Together, we have the complete Reynolds stress model:

$$\langle u_i' u_j' \rangle = \underbrace{\frac{2}{3}k\delta_{ij}}_{\text{Isotropic}} + \underbrace{2kb_{ij}}_{\text{Anisotropic}}$$

Here k controls the overall turbulence levels, while b_{ij} captures directional effects (e.g., $\langle u'^2 \rangle \neq \langle v'^2 \rangle$ in shear flows).

An exmaple for this would be the Boussinesq approximation, where

$$b_{ij} \approx -\nu_t \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right)$$

Quantity	**Role**	**Equation**
k	Amplitude of fluctuations	$\frac{1}{2}\langle u_i' u_i' \rangle$
b_{ij}	Anisotropy shape	$\frac{\langle u_i' u_j' \rangle}{2k} - \frac{1}{3}\delta_{ij}$
ν_t	Turbulent diffusivity	$\nu_t \sim k^{1/2} L$

where $\nu_t = k^{1/2} L$ is the turbulent viscosity with L as a length scale.

(20250203#94)

From the material derivative of velocity $\boldsymbol{u}$, obtain the expression for material derivative for kinetic energy.

We have,

$$\frac{D\mathbf{u}}{Dt} = \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u}$$

Taking dot product with velocity,

$$\mathbf{u} \cdot \frac{D\mathbf{u}}{Dt} = \mathbf{u} \cdot \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot [(\mathbf{u} \cdot \nabla)\mathbf{u}]$$

Unsteady acceleration term would be

$$\mathbf{u} \cdot \frac{\partial \mathbf{u}}{\partial t} = \frac{1}{2}\frac{\partial}{\partial t}(\mathbf{u} \cdot \mathbf{u}) = \frac{\partial}{\partial t}\left(\frac{|\mathbf{u}|^2}{2}\right)$$

while non-linear advection term would be

$$\boldsymbol{u}\left[(\boldsymbol{u} \cdot \nabla \boldsymbol{u})\right] = u_j \left(u_i \frac{\partial u_j}{\partial x_i}\right) = u_i u_j \frac{\partial u_j}{\partial x_i}$$

Using the product rule,

$$\mathbf{u} \cdot [(\mathbf{u} \cdot \nabla)\mathbf{u}] = \mathbf{u} \cdot \nabla \left(\frac{|\mathbf{u}|^2}{2}\right)$$

Final simplified form

$$\mathbf{u} \cdot \frac{D\mathbf{u}}{Dt} = \frac{\partial}{\partial t}\left(\frac{|\boldsymbol{u}|^2}{2}\right) + \boldsymbol{u} \cdot \nabla \left(\frac{|\boldsymbol{u}|^2}{2}\right) = \frac{D}{Dt}\left(\frac{|\boldsymbol{u}|^2}{2}\right)$$

$D/Dt\,(|\boldsymbol{u}|^2/2)$ describes how the kinetic energy of the fluid parcel changes as it moves. In conservative flows, with no viscosity or external forces, this would equate directly to work done by pressure forces.

(20250203#95)

Derive the energy equation for a flow with no external forcing, and explain what happens to the kinetic energy of the flow:

Momentum equation in vector form

$$\rho \frac{D\mathbf{u}}{Dt} = \nabla \cdot \tau$$

where τ is the total stress tensor (pressure + viscous stresses).

In indicial notation, it becomes

$$\rho \frac{Du_i}{Dt} = \frac{\partial \tau_{ji}}{\partial x_j}$$

which can be interpreted as the acceleration in the i-th direction as a result of forcing in the i-th direction.

Multiply both LHS nad RHS by u_i,

$$\rho u_i \frac{Du_i}{Dt} = u_i \frac{\partial \tau_{ji}}{\partial x_j}$$

Rewrite LHS,

$$\rho u_i \frac{Du_i}{Dt} = \rho \frac{D}{Dt} \left(\frac{1}{2} u_i u_i \right)$$

The velocity gradient can be decomposed as symmetric (strain rate S_{ij}) and antisymmetric (rotation rate Ω_{ij}) parts:

$$S_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right), \quad \Omega_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right)$$

giving

$$\frac{\partial u_i}{\partial x_j} = S_{ij} + \Omega_{ij}$$

Stress tensor for Newtonian fluid is

$$\tau_{ji} = -P\delta_{ji} + 2\mu S_{ji}$$

consisting of isotropic pressure and viscous stress terms.

Consider the product $\tau_{ji} \partial u_i / \partial x_j$. We have

$$\tau_{ji} \frac{\partial u_i}{\partial x_j} = \left(-P\delta_{ji} + 2\mu u S_{ji} \right) \left(S_{ij} + \Omega_{ij} \right)$$

Pressureu term vanishes as

$$-P\delta_{ji} S_{ij} = -P S_{ii} = -P \nabla \cdot \boldsymbol{u} = 0$$

as $\nabla \cdot \boldsymbol{u} = 0$ for an incompressible flow.

The viscous term would be $2\mu S_{ji}S_{ij}$ as $S_{ji}\Omega_{ij}$ would vanish due to symmetry/antisymmetry. Thus, for an incompressible flow,

$$\tau_{ji}\frac{\partial u_i}{\partial x_j} = 2\mu S_{ji}S_{ij}$$

Substituting this back into the energy equation,

$$\rho\frac{D}{Dt}\left(\frac{1}{2}u_i u_i\right) = \frac{\partial}{\partial x_j}(u_i\tau_{ji}) - \tau_{ji}\frac{\partial u_i}{\partial x_j}$$

We have,

$$\frac{\partial}{\partial t}\left(\frac{1}{2}\rho u_i u_i\right) + \frac{\partial}{\partial x_j}\left(\rho u_j\left(\frac{1}{2}u_i u_i\right)\right) = \frac{\partial}{\partial x_j}(u_i\tau_{ji}) - 2\mu S_{ji}S_{ij}$$

$$\rho\frac{D}{Dt}\left(\frac{1}{2}u_i u_i\right) = \frac{\partial}{\partial x_j}(u_i\tau_{ji}) - 2\mu S_{ji}S_{ij}$$

$$\rho\frac{DE}{Dt} + \rho\left(\nabla\cdot T\right) = -2\rho\nu S_{ji}S_{ij},$$

where $T_i = u_i P/\rho - 2\nu u_i S_{ji}$. The equation above describes the rate at which energy carried by a fluid parcel changes due to viscous and a transport term.

For a periodic box, integrate E over the spatial domain to get the total energy carried within the volume. If we look at how E varies with time, under no forcing, we observe that T vanishes and we are only left with $-2\rho\nu S_{ji}S_{ij}$. Thus in the absence of forcing, kinetic energy of the flow keeps decaying due to viscosity.

(20250207#96)

Use brute force approach to derive equation for Reynolds stress tensor and explain how that's not a good idea:

Reynold's stress tensor is defined as

$$\tau_{ij}^R = -\rho\langle u_i' u_j'\rangle$$

To model this, we can use

$$\langle u_i' u_j'\rangle = F(u_i;\ \text{class of flows})$$

Instead of modelling, we can solve for τ^R using brute force approach. To do so, subtract RANS from instantaneous Navier-Stokes equation,

$$\frac{\partial u_i}{\partial t} + u_j\frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \nu\nabla^2 u_i$$

Subtracting the RANS equation,

$$\frac{\partial U_i}{\partial t} + U_j\frac{\partial U_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial P}{\partial x_i} + \nu\nabla^2 U_i - \frac{\partial\langle u_i' u_j'\rangle}{\partial x_j}$$

Resulting in the equation for fluctuations u_i':

$$\frac{\partial u_i'}{\partial t} + U_k \frac{\partial u_i'}{\partial x_k} + u_k' \frac{\partial U_i}{\partial x_k} + \frac{\partial}{\partial x_k}(u_i' u_k' - \langle u_i' u_k' \rangle) = -\frac{1}{\rho}\frac{\partial p'}{\partial x_i} + \nu \nabla^2 u_i'$$

To derive the transport equation for $\langle u_i' u_j' \rangle$, multiply the fluctuation equation by u_j' and average to get:

$$\frac{D\langle u_i' u_j' \rangle}{Dt} = P_{ij} + \Pi_{ij} - \epsilon_{ij} + \frac{\partial}{\partial x_k}\left(\nu \frac{\partial \langle u_i' u_j' \rangle}{\partial x_k} + C_{ijk}\right)$$

where P_{ij} is the production term given by

$$P_{ij} = -\langle u_i' u_k' \rangle \frac{\partial U_j}{\partial x_k} - \langle u_j' u_k' \rangle \frac{\partial U_i}{\partial x_k}$$

Π_{ij} is the pressure-strain term given by

$$\Pi_{ij} = \left\langle p' \left(\frac{\partial u_i'}{\partial x_j} + \frac{\partial u_j'}{\partial x_i} \right) \right\rangle$$

ϵ_{ij} is the dissipation term expressed as

$$\epsilon_{ij} = 2\nu \left\langle \frac{\partial u_i'}{\partial x_k} \frac{\partial u_j'}{\partial x_k} \right\rangle$$

and C_{ijk} is the triple correlation term whose expression is

$$C_{ijk} = \langle u_i' u_j' u_k' \rangle + \frac{\langle p' u_i' \rangle}{\rho}\delta_{jk} + \frac{\langle p' u_j' \rangle}{\rho}\delta_{ik}$$

The triple correlation term introduces higher order moments. We can go further and derive the evolution equation for this triple correlation as well, but it introduces even higher moments and this process keeps on going with us ending up with higher and higher order moments and an infinite hierarchy. Thus we are unable to close the equation. This is called turbulence closure problem. To close the equation, we model out thew unkonwn terms rather than following a brute force approach.

(20250207#97)

Is small scale energy in turbulent flows negligible?

If small-scale energy were truly negligible, we could truncate the hierarchy—but real turbulence exhibits intermittency, making this approximation invalid. Hence, modeling is essential.

Note: Intermittency in turbulence refers to the irregular and sporadic occurrence of intense velocity fluctuations, particularly at small scales. This phenomenon is directly related to the energy cascade and the way turbulence dissipates energy.

(20250207#98)

State and explain turbulent viscosity hypothesis. Substitute the obtained expression Reynolds stress tensor into the governing equations:

The Reynolds stress tensor is modeled analogously to viscous stress in Newtonian fluids:

$$\tau_{ij}^{\text{turb}} = -\rho\langle u_i' u_j' \rangle = \rho\nu_t \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) - \frac{2}{3}\rho k \delta_{ij}$$

Property	**Molecular Viscosity (ν)**	**Turbulent Viscosity (ν_t)**
Nature	Fluid property (constant)	Flow property (varies)
Expression	$\nu = \mu/\rho$	$\nu_t = f(\text{flow conditions})$
Dependence	Temperature, pressure	Velocity gradients, turbulence scale

Substituting the turbulent viscosity hypothesis into RANS:

$$\frac{\overline{D}U_i}{Dt} = \frac{\partial}{\partial x_j}\left[\nu_{\text{eff}}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right)\right] - \frac{1}{\rho}\frac{\partial}{\partial x_i}\left(P + \frac{2}{3}\rho k\right)$$

Using incompressibility condition,

$$\nabla \cdot \mathbf{U} = 0 \quad \text{or} \quad \frac{\partial U_i}{\partial x_i} = 0$$

Effective viscosity

$$\nu_{\text{eff}} = \nu + \nu_t(x, t)$$

$$\nu_t = \nu_t(x) \quad \text{(stationary flow)}$$

(20250207#99)

What is the kinetic theory analog for ν_t? What are some other ways of determinin ν_t?

Using Prandtl's mixing length hypothesis (momentum transport by turbulent eddies):

$$\nu_t = l_m^2 \left| \frac{\partial U}{\partial y} \right|$$

where l_m is the mixing length. For a boundary layer,

$$\nu_t = \kappa y u_\tau \quad \text{(log-law region)}$$

Complete system of equations would be

$$\frac{\partial U_i}{\partial t} + U_j \frac{\partial U_i}{\partial x_j} = \frac{\partial}{\partial x_j}\left[(\nu + \nu_t)\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right)\right] - \frac{1}{\rho}\frac{\partial P^*}{\partial x_i}$$

where $P^* = P + 2\rho k/3$ and

$$\frac{\partial U_i}{\partial x_i} = 0$$

Commonly used turbulence models to come up with ν_t:

- Prandtl's mixing length
$$\nu_t = l_m^2 \left|\frac{dU}{dy}\right|$$
- $k - \epsilon$ model
$$\nu_t = C_\mu \frac{k^2}{\epsilon}$$
- $k - \omega$ model
$$\nu_t = \frac{k}{\omega}$$

(20250207#100)

What would be a model for ν_t for a turbulent round jet?

The turbulent viscosity ν_t for a round jet is modeled as a function of centerline velocity $U_c(x)$, jet half-width $r_{1/2}(x)$, and similarity variable $\eta = r/r_{1/2}$. General formulation: '

$$\nu_t(x, r) = U_c(x) \cdot r_{1/2}(x) \cdot \hat{\nu_T}(\eta)$$

One such simplified model for practical calculations will have $\hat{\nu}_T$ approximated as

$$\hat{\nu_T}(\eta) = \begin{cases} \nu_0 & \text{for } \eta \leq 1 \\ 0 & \text{for } \eta > 1 \end{cases}$$

where ν_0 is determined experimentally. The centerline velocity decay is of the form $U_c(x) \propto 1/x$, jet spreading rate variation in downstream direction is $r_{1/2}(x) \propto x$, so the complete model will have dependence on x as

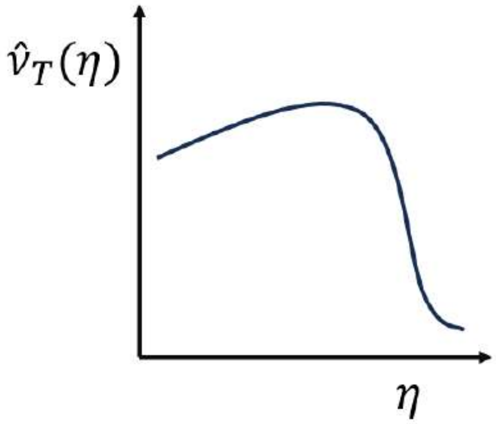
$$\nu_t(x) = \frac{k}{x} \cdot x \cdot \nu_0 = k\nu_0 \quad \text{(constant)}$$

The constant ν_0 is typically found from experiments to be

$$\nu_0 \approx 0.025 \pm 0.002$$

When used in RANS equations, we'll end up with

$$\frac{\partial U}{\partial t} + U\frac{\partial U}{\partial x} + V\frac{\partial U}{\partial r} = \frac{1}{r}\frac{\partial}{\partial r}\left(r(\nu + \nu_t)\frac{\partial U}{\partial r}\right)$$



Feature	Mathematical Expression
Turbulent viscosity	$\nu_t(x,r) = U_c(x) r_{1/2}(x) \hat{\nu_T}(\eta)$
Similarity variable	$\eta = r/r_{1/2}$
Simplified model	$\hat{\nu_T}(\eta) = \begin{cases} \nu_0 & \eta \leq 1 \\ 0 & \eta > 1 \end{cases}$
Centerline decay	$U_c(x) \sim x^{-1}$
Jet spreading	$r_{1/2}(x) \sim x$

(20250210#101)

(20250214#102)

Express turbulent viscosity for $k - \omega$ and $k - \epsilon$ models:

- **Two-equation turbulence models** aim to provide a more accurate closure of the Reynolds-Averaged Navier–Stokes (RANS) equations by solving transport equations for two turbulence quantities:
  - k: Turbulent kinetic energy
  - ε or ω: Turbulent dissipation rate or specific dissipation rate
- Examples include:
  - k-ε model
  - k-ω model
- These models compute the turbulent viscosity ν_T from the solved quantities, e.g.,

$$\nu_T \sim \frac{k^2}{\varepsilon} \quad (\text{for } k\text{-}\varepsilon), \quad \nu_T \sim \frac{k}{\omega} \quad (\text{for } k\text{-}\omega)$$

- The friction velocity u_τ, defined as:

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}}$$

  is often computed from the wall shear stress τ_w and passed as an input or used in wall functions to model near-wall behavior.

(20250214#103)

How to distribute grid points to capture the near-wall velocity profile properly?

Discretization and wall function capture:

- Accurate representation of the near-wall velocity profile $u^+(y^+)$ requires that the grid is refined enough in wall units:

$$y^+ = \frac{y u_\tau}{\nu}$$

- In the viscous sublayer:

$$u^+ \propto y^+ \quad \Rightarrow \text{Linear relation; two points may be enough}$$

- In the logarithmic layer:

$$u^+ \propto \log(y^+) \quad \Rightarrow \text{Nonlinear; requires more than two points to resolve accurately}$$

Therefore, grid resolution near the wall must be chosen based on the region being resolved:

- If the first few grid points lie in the viscous sublayer, coarse discretization may suffice.
- If they lie in the buffer or logarithmic layer, finer resolution is necessary to capture the curvature of the logarithmic profile.

This understanding is essential when implementing wall functions and evaluating turbulence model accuracy in capturing boundary layer dynamics.

(20250214#104)

Explain geometric progression for grid spacing in the case of non-uniform grid clustering:

- In wall-bounded turbulent flows, such as channel or pipe flow, accurately capturing the steep gradients near the wall is essential.
- If a **uniform grid** is used across the entire domain, the spacing near the wall may be too coarse to resolve the boundary layer, particularly in the viscous sublayer where high velocity gradients exist.
- **Solution: Grid Clustering**
  - Use **non-uniform grid spacing** by clustering points near the wall and using coarser spacing away from it.
  - This allows better resolution where it is most needed (near the wall) without excessively increasing the total number of grid points.
- **Geometric Progression for Grid Spacing**
  - One practical approach is to design the spacing between adjacent grid points to follow a geometric progression:
    $$\frac{\Delta^{k+1}}{\Delta^k} = r$$
    where Δ^k is the spacing between the k-th and $(k+1)$-th points, and r is the common ratio.
  - A typical value that works well in practice is:
    $$r \approx 1.02$$
    meaning that the grid spacing grows by about 2% with each step away from the wall.
- **Advantages of Geometrically Clustered Grids**:
  - High resolution near the wall for accurate representation of steep gradients and sublayer structures.

  - Reduced computational cost compared to using uniformly fine grids throughout the domain.
  - Ensures smoother transition in spacing which benefits numerical stability and accuracy.

- **Example Application**:
  - Grid points are clustered near $y = 0$ and stretched gradually into the bulk region of the flow.
  - This method is particularly useful for resolving $u^+(y^+)$ profiles or capturing near-wall turbulence behavior in LES and DNS.

(20250214#105)

What is the effect of pressure gradient on the boundary layer and how to handle it in the mesh?

- In boundary layer flows, the pressure gradient along the wall significantly affects the thickness of the boundary layer and the required resolution for accurate simulation.
- **Favorable Pressure Gradient (FPG)**:
  - Occurs when the pressure decreases in the direction of the flow:

$$\frac{dP}{dx} < 0$$

  - The accelerating flow helps suppress separation and reduces the boundary layer thickness.
  - As a result, the velocity gradient near the wall increases, and finer grid resolution is needed in the wall-normal direction to accurately capture the sharper profile.
  - **Implication:** Cluster grid points more tightly near the wall to resolve the thinner boundary layer.
- **Adverse Pressure Gradient (APG)**:
  - Occurs when the pressure increases in the direction of the flow:

$$\frac{dP}{dx} > 0$$

  - The decelerating flow causes the boundary layer to grow thicker and may lead to flow separation if the gradient is strong enough.
  - The wall-normal gradients become weaker compared to FPG cases, but the outer region of the boundary layer becomes more significant.
  - **Implication:** Grid clustering should be more spread out across a thicker region of the boundary layer, with points not as tightly concentrated only at the wall.

- **Clustering Strategy Based on Pressure Gradient**:
  - In regions with favorable pressure gradient:

    $$\text{Use fine clustering near the wall (e.g., geometric spacing with } r \approx 1.02\text{)}$$

  - In regions with adverse pressure gradient:

    $$\text{Spread grid points over a larger wall-normal extent to resolve the thicker boundary layer}$$

  - **Adaptive meshing** or a combination of grid types may be required for simulations involving both FPG and APG.

(20250214#106)

Why does the boundary layer become thicker and thinner based on adverse and favorable pressure gradient?

- Consider the boundary layer developing over a flat or curved surface. The growth of the boundary layer is governed by the momentum balance near the wall, particularly the interplay between the pressure gradient and the inertial and viscous forces.
- The streamwise momentum equation in the boundary layer (incompressible) is:

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = -\frac{1}{\rho}\frac{dP}{dx} + \nu\frac{\partial^2 u}{\partial y^2}$$

  where $\frac{dP}{dx}$ is the imposed pressure gradient from the outer (inviscid) flow.
- **Favorable Pressure Gradient (FPG)**:
  - Defined by $\frac{dP}{dx} < 0$, i.e., pressure decreases along the flow direction.
  - This accelerates the flow within the boundary layer, adding energy to the fluid particles near the wall.
  - Increased momentum near the wall resists the viscous retardation, leading to a **steeper velocity profile** and hence a **thinner boundary layer**.
  - Vorticity is compressed toward the wall, and flow separation is suppressed.
- **Adverse Pressure Gradient (APG)**:
  - Defined by $\frac{dP}{dx} > 0$, i.e., pressure increases along the flow direction.
  - This decelerates the flow and removes kinetic energy from the fluid particles near the wall.
  - Near-wall fluid loses momentum and struggles to overcome viscous resistance, leading to a **flattened velocity profile** and hence a **thicker boundary layer**.
  - If the pressure gradient is strong enough, the velocity near the wall can reverse direction, causing **flow separation**.

- **Summary of Effects**:
  - **FPG:** Velocity increases, boundary layer thins, and flow remains attached.
  - **APG:** Velocity decreases, boundary layer thickens, and there is risk of separation.

(20250214#107)

Where does the minimum grid spacing occur in a wall-bounded flow and how much is it?

- In simulations of wall-bounded flows (DNS, LES, RANS), accurate resolution near the wall is essential.
- The wall-normal spacing of the first grid point from the wall is often expressed in wall units as:
  $$\Delta_1^+ = \frac{\Delta_1 \, u_\tau}{\nu}$$
  where:
  - Δ_1 is the physical distance of the first grid point from the wall,
  - $u_\tau = \sqrt{\tau_w/\rho}$ is the friction velocity,
  - ν is the kinematic viscosity.
- The choice of Δ_1^+ determines the ability to resolve near-wall structures like the viscous sublayer ($y^+ \lesssim 5$):
  - For DNS: $\Delta_1^+ \lesssim 1$,
  - For LES: $\Delta_1^+ \sim 1$ or slightly larger,
  - For RANS: wall functions may allow larger values.
- When setting up the computational grid, we **estimate** Δ_1^+ using an **assumed value of** u_τ based on empirical correlations or prior simulations.
- Since u_τ is unknown until the solution is obtained, the actual Δ_1^+ is validated after the simulation by computing:
  $$\Delta_1^+ = \frac{\Delta_1 \, u_\tau^{\text{computed}}}{\nu}$$
- Often, when someone says "$y^+ = 1$", they mean:
  - The first grid point from the wall lies at $\Delta_1^+ = 1$,
  - i.e., the grid spacing in wall units is unity or less, sufficient to resolve the viscous sublayer.
- The relationship between Reynolds numbers based on channel height and friction velocity (e.g., $Re_\tau = \frac{u_\tau \delta}{\nu}$) can be used to estimate u_τ from bulk quantities to precompute Δ_1 for a target Δ_1^+.

(20250214#108)

Give some practical considerations which come into picture when we try to come up with grid resolution of different physical flow scenarios:

- In bluff body flows (e.g., flow over a cylinder), the wake exhibits vortex shedding at a characteristic frequency f. The non-dimensional shedding frequency is captured by the **Strouhal number**:

$$St = \frac{fD}{U}$$

  where:
  - f: vortex shedding frequency,
  - D: characteristic length (e.g., diameter),
  - U: free stream velocity.

- Experimental data often shows $St \sim 0.2$ for flow over a circular cylinder at moderate Reynolds numbers. Numerical computations aim to reproduce this value accurately.

- Apart from unsteady dynamics, we are also interested in integrated quantities such as **total drag force**. This drag force consists of:
  - **Friction drag**, which depends on the distribution of u_τ (friction velocity) over the surface,
  - **Pressure drag**, which depends on the surface pressure distribution.

- Obtaining accurate u_τ values numerically is difficult, especially in turbulent flows, due to:
  - Steep velocity gradients near the wall,
  - Requirement for fine resolution near the wall to resolve the viscous sublayer.

- **Heat transfer problems** are even more challenging. This is because the wall heat flux depends directly on the **temperature gradient** at the wall:

$$q'' = -k \left. \frac{\partial T}{\partial y} \right|_{\text{wall}}$$

  Accurately computing $\partial T/\partial y$ at the wall requires proper resolution of the thermal boundary layer.

- In turbulent flows, temperature distribution is strongly influenced by turbulent transport. The structure and intensity of these fluctuations govern scalar transport.

- The **Prandtl number** $\mathrm{Pr} = \frac{\nu}{\alpha}$ characterizes the relative thicknesses of the velocity and thermal boundary layers:
  - For $\mathrm{Pr} = 1$: Thermal and momentum boundary layers have similar thickness,
  - For $\mathrm{Pr} \neq 1$: They differ in thickness, and the thinner boundary layer must be resolved for accurate results.

- **Resolution of the thinner boundary layer** is essential:
  - If the thermal boundary layer is thinner (e.g., $\mathrm{Pr} > 1$), it must be resolved to compute heat transfer accurately.

  - If neither boundary layer is resolved, the computed solution will fail to capture correct viscous forces or conductive heat fluxes near the wall.

(20250214#109)

Comapre $k - \omega$ and $k - \epsilon$ models:

- The k-ω and k-ϵ models are two of the most commonly used two-equation turbulence models for Reynolds-averaged Navier–Stokes (RANS) simulations. Each model has its own advantages and limitations based on the flow configuration and near-wall treatment.
- **k-ω model:**
  - Less sensitive to near-wall grid spacing compared to the k-ϵ model.
  - Performs well in resolving the flow close to the wall, even with a coarser or stretched mesh.
  - Allows **fewer grid points** in the boundary layer region while maintaining good accuracy, especially in capturing wall shear stress and boundary layer development.
  - **Advantage:** Suitable for wall-bounded flows with steep gradients and wall shear stress predictions.
  - **Limitation:** Tends to be overly sensitive to **free-stream turbulence fluctuations**. This can result in incorrect turbulence generation or dissipation in the outer flow, especially in external aerodynamics.
- **k-ϵ model:**
  - More robust in the free-stream region and less sensitive to external turbulence levels.
  - Requires careful near-wall treatment, particularly the correct specification of the first cell height to ensure appropriate y^+ values.
  - **Limitation:** Sensitive to y^+, the dimensionless wall distance:

$$y^+ = \frac{yu_\tau}{\nu}$$

  - For wall-function approaches, y^+ should typically lie in the range $30 < y^+ < 300$.
  - If used in low Reynolds number form (resolving down to the viscous sublayer), it requires very fine mesh with $y^+ \lesssim 1$, which increases computational cost.

(20250214#110)

Explain SST model:

- In real-world applications, neither the k-ω nor the k-ϵ model is universally superior across all regions of the flow domain.

- Each model performs better in different regions:
  - **k-ω:** Accurate in the near-wall region, but overly sensitive to free-stream turbulence fluctuations.
  - **k-ϵ:** More robust in the free-stream region, but sensitive to near-wall grid resolution, especially the minimum y^+.

- To take advantage of the strengths of both models, the **Shear Stress Transport (SST)** model was developed.

- SST uses a **blending function** $F_1(y) \in [0, 1]$ to smoothly transition between the two models:

$$\text{Near wall:} \quad F_1 \approx 1 \Rightarrow \text{use } k\text{-}\omega \quad (\text{low sensitivity to } y^+)$$

$$\text{Far from wall:} \quad F_1 \approx 0 \Rightarrow \text{use } k\text{-}\epsilon \quad (\text{robust to free-stream})$$

- The turbulent kinetic energy equation (for k) is the same throughout and consistent with both models. The difference arises in the dissipation term:

$$\nu_T = \frac{k}{\omega} \quad (\text{as in the } k\text{-}\omega \text{ model})$$

- **Key advantages of the SST model:**
  - Accurate near-wall treatment without requiring extremely fine grids.
  - Reduced sensitivity to free-stream turbulence properties.
  - Improved prediction of adverse pressure gradient flows and flow separation.

- In summary, SST is a **hybrid turbulence model** that blends the k-ω and k-ϵ models using a spatially-varying function, thereby improving overall robustness and accuracy.

(20250214#111)

How does the two-equation models handle internal and external flows?

- In the context of turbulent flow modeling, the choice between the k-ω and k-ϵ models can significantly influence the accuracy of simulations, especially in external flows.

- However, for **internal flows**—such as flow through pipes, ducts, or channels—the distinction between the two models becomes less critical.

- This is primarily because internal flows typically lack a substantial **free-stream region**, unlike external flows where large ambient regions (e.g., flow over an airfoil or fuselage) are present.

- In external flows:

  - The k-ω model tends to be overly sensitive to small free-stream turbulence levels.
  - The k-ϵ model is more stable in these regions but struggles near the wall due to its dependence on a proper y^+ resolution.

- In internal flows:
  - Both the k-ω and k-ϵ models operate in similar near-wall-dominated environments.
  - Since there is minimal or no free-stream influence, the specific sensitivities of each model to wall proximity or ambient turbulence are less pronounced.

- **Conclusion:** For internal flows, either model can be employed with comparable effectiveness, assuming proper near-wall treatment and sufficient grid resolution. The choice may then depend more on implementation convenience or legacy code preferences rather than significant differences in predictive capability.

(20250214#112)

How well do the two equation RANS models perform under off-design and transition conditions?

- In **high Reynolds number** (Re) flows, the transition from laminar to turbulent flow occurs very rapidly, often very close to the leading edge of a surface.

- Due to this rapid transition, it is generally acceptable to apply turbulence models such as k-ϵ or k-ω from the leading edge onward. This assumption is justified because:
  - The flow is fully turbulent over most of the domain.
  - The influence of the transition zone is small or negligible.

- However, under **off-design conditions**, the Reynolds number may not remain high. As a result:
  - The flow may transition from laminar to turbulent further downstream.
  - Applying a turbulence model such as k-ϵ or k-ω starting at the leading edge would incorrectly assume the flow is already turbulent there.
  - This can lead to **large modeling errors** in the prediction of separation, heat transfer, and drag.

- **Example:** Consider the flow over a turbine blade operating at off-design conditions:
  - The Reynolds number may be lower than expected.
  - The boundary layer near the leading edge may still be laminar.
  - If a k-ω or k-ϵ model is used from the leading edge onward, the model would artificially impose turbulence.
  - This results in **incorrect predictions** of boundary layer growth, transition location, and related aerodynamic quantities.

- **Conclusion:** Care must be taken when choosing the location to initiate turbulence models, particularly in off-design simulations. Accurate transition modeling or hybrid methods (e.g., transition-sensitive models or intermittency-based models) are required to ensure physical accuracy in such scenarios.

(20250214#113)

Why use k, ϵ and ω in turbulence modelling?

- **Turbulent Kinetic Energy (k) as Fundamental Invariant**:

$$k \equiv \frac{1}{2}\overline{u_i' u_i'}$$

  - Represents the amplitude of velocity fluctuations
  - First invariant of Reynolds stress tensor
  - Direct measure of turbulence intensity

- **Necessity for Two-Equation Models**:
  - Single equation (just k) insufficient - missing length scale
  - Velocity scale: $v \sim \sqrt{k}$
  - Length scale required for complete turbulence characterization

The ϵ vs ω Consideration

- **Dissipation Rate (ϵ) Approach**:

$$\epsilon \equiv \nu \overline{\frac{\partial u_i'}{\partial x_k}} \frac{\partial u_i'}{\partial x_k}$$

  - Traditional choice for second variable (k-ϵ models)
  - Provides turbulence length scale via $L \sim k^{3/2}/\epsilon$

- **Specific Dissipation (ω) Alternative**:

$$\omega \equiv \frac{\epsilon}{k}$$

  - Dimensionally equivalent to frequency (time^{-1})
  - Directly relates dissipation to kinetic energy
  - More numerically stable than ϵ formulation

Advantages of k-ω Formulation

- **Enhanced Numerical Stability**:
  - ω equations better conditioned near walls
  - Avoids singularities in viscous sublayer
  - More robust for adverse pressure gradients
- **Physical Interpretation**:

$$\text{Eddy viscosity} \quad \nu_T = \frac{k}{\omega}$$

  - Natural formulation for eddy viscosity
  - Automatically enforces realizability
- **Empirical Performance**:
  - Better prediction of separation points
  - Accurate for boundary layers with pressure gradients
  - Works well for transitional flows

Comparison with k-ϵ Models

- **Wall Treatment**:
  - k-ω: Integrates to wall without damping functions
  - k-ϵ: Requires wall functions or low-Re modifications
- **Free Shear Flows**:
  - k-ω: May overpredict spreading rates
  - Solution: Use SST (Shear Stress Transport) variant

Mathematical Formulation

- **Standard k-ω Equations**:

$$\frac{\partial k}{\partial t} + U_j \frac{\partial k}{\partial x_j} = P_k - \beta^* k\omega + \frac{\partial}{\partial x_j}\left[(\nu + \sigma_k \nu_T)\frac{\partial k}{\partial x_j}\right]$$

$$\frac{\partial \omega}{\partial t} + U_j \frac{\partial \omega}{\partial x_j} = \alpha \frac{\omega}{k} P_k - \beta\omega^2 + \frac{\partial}{\partial x_j}\left[(\nu + \sigma_\omega \nu_T)\frac{\partial \omega}{\partial x_j}\right]$$

where P_k is production term and β^*, α, β are model constants.

(20250214#114)

Of $k - \epsilon$ and $k - \omega$ models, which one performs better in the presence of favorable pressure gradient?

- The standard k-ϵ turbulence model demonstrates superior performance compared to k-ω models in flows with strong adverse pressure gradients. This advantage stems from several key factors:
  1. The ϵ transport equation contains production-to-dissipation ratio terms that better capture the turbulence modification under pressure gradient effects
  2. The model constants in k-ϵ are calibrated for equilibrium boundary layers, which frequently experience pressure gradients
  3. The turbulent viscosity formulation $\nu_t = C_\mu k^2/\epsilon$ provides more stable predictions when flow separation occurs due to pressure gradients

- However, standard k-ϵ requires modifications for accurate predictions:
  - Realizable k-ϵ variants constrain normal stresses to remain positive
  - RNG k-ϵ adds an analytical differential term for strain effects
  - Near-wall treatment still remains challenging without proper damping functions

Anisotropy of Reynolds Stress Terms

- The fundamental limitation of both k-ϵ and k-ω models lies in their treatment of Reynolds stress anisotropy:
  - They assume isotropic eddy viscosity: $\overline{u_i'u_j'} = \frac{2}{3}k\delta_{ij} - \nu_t\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right)$
  - This Boussinesq approximation fails in:
    1. Strongly curved flows
    2. Rapidly accelerating/decelerating flows
    3. Flows with strong rotation
    4. Near-wall regions with normal stress anisotropy

- Advanced modeling approaches for anisotropy:
  - Explicit Algebraic Stress Models (EASM) that include nonlinear terms
  - Full Reynolds Stress Models (RSM) solving transport equations for each $\overline{u_i'u_j'}$ component
  - Anisotropy-resolving LES/DNS for fundamental studies

(20250214#115)

Give the fundamental equation for Reynold's stress transport and explain the terms involved:

- The fundamental equation for Reynolds stress transport is given by:

$$\frac{\overline{D}}{Dt}\langle u_i'u_j'\rangle + \frac{\partial}{\partial X_k}T_{kij} = \mathcal{P}_{ij} + \mathcal{R}_{ij} - \epsilon_{ij} \tag{10}$$

  - Where $\langle u_i' u_j' \rangle$ represents the Reynolds stress tensor components

- **Left-Hand Side (LHS) Terms**:
  - First term: Material derivative of Reynolds stresses

$$\frac{\overline{D}}{Dt}\langle u_i' u_j' \rangle = \frac{\partial}{\partial t}\langle u_i' u_j' \rangle + U_k \frac{\partial}{\partial X_k}\langle u_i' u_j' \rangle \tag{11}$$

    * Represents the total rate of change of $\langle u_i' u_j' \rangle$ carried by the mean flow
    * Contains both local time derivative and convective transport
  - Second term: Spatial transport term

$$\frac{\partial}{\partial X_k}T_{kij} = \frac{\partial}{\partial X_k}\left[\langle u_i' u_j' u_k' \rangle + \frac{1}{\rho}\langle p' u_i' \rangle \delta_{jk} + \frac{1}{\rho}\langle p' u_j' \rangle \delta_{ik} - \nu \frac{\partial}{\partial X_k}\langle u_i' u_j' \rangle\right] \tag{12}$$

    * Represents the flux of Reynolds stresses across boundaries
    * For a finite control volume, this gives the net efflux of $\langle u_i' u_j' \rangle$
    * Contains turbulent diffusion, pressure diffusion, and viscous diffusion components

- **Right-Hand Side (RHS) Terms**:
  - First term: Production tensor $\mathcal{P}_{ij}$

$$\mathcal{P}_{ij} = -\langle u_i' u_k' \rangle \frac{\partial U_j}{\partial X_k} - \langle u_j' u_k' \rangle \frac{\partial U_i}{\partial X_k} \tag{13}$$

    * Represents production of Reynolds stresses by mean velocity gradients
    * Mechanism of energy transfer from mean flow to turbulence
    * Does not change total turbulent kinetic energy, just redistributes it among components
  - Second term: Pressure-rate-of-strain tensor $\mathcal{R}_{ij}$

$$\mathcal{R}_{ij} = \left\langle \frac{p'}{\rho}\left(\frac{\partial u_i'}{\partial X_j} + \frac{\partial u_j'}{\partial X_i}\right)\right\rangle \tag{14}$$

    * Accounts for redistribution of turbulent energy among stress components
    * Results from interaction between fluctuating pressure and fluctuating strain rate
    * Tends to make turbulence more isotropic
  - Third term: Dissipation tensor ϵ_{ij}

$$\epsilon_{ij} = 2\nu \left\langle \frac{\partial u_i'}{\partial X_k}\frac{\partial u_j'}{\partial X_k}\right\rangle \tag{15}$$

    * Represents viscous dissipation of velocity fluctuation correlations
    * For isotropic turbulence: $\epsilon_{ij} = \frac{2}{3}\epsilon\delta_{ij}$
    * In general flows, the dissipation tensor is anisotropic

(20250214#116)

How is RANS dependence on the model used make it less consistent as compared to other numerical methods?

- The Reynolds-Averaged Navier-Stokes (RANS) approach exhibits significant dependence on the specific turbulence model employed:
    - Different RANS models can predict fundamentally different flow behaviors for the same physical configuration
    - A critical example appears in compressor blade wake flows:
        * Some models may predict only a separation bubble
        * Others may predict complete separation with downstream reattachment
        * These discrepancies arise from varying closure approximations
    - The variability stems from how each model handles:
        * Turbulent stress anisotropy
        * Pressure-strain correlations
        * Near-wall turbulence effects
- In contrast, Large Eddy Simulation (LES) demonstrates more consistent behavior:
    - LES resolves the large, energy-containing eddies directly
    - Only small scales are modeled, reducing overall dependence on closure approximations
    - Provides more reliable predictions for complex flows like:
        * Separated flows
        * Vortex shedding
        * Transitional boundary layers

(20250214#117)

How is eddy viscosity determined in one equation vs two equation models?

Eddy Viscosity Determination in Turbulence Models

- The primary purpose of turbulence models is to determine the eddy viscosity ν_T:

$$\nu_T = \text{function of turbulence variables} \tag{16}$$

- **Single-Equation Models (e.g., k-equation)**:
    - Solve only for turbulent kinetic energy k

$$\frac{Dk}{Dt} = P_k - \epsilon + \text{transport terms} \tag{17}$$

  - Require external specification of a length scale L:

$$\nu_T = C_\mu k^{1/2} L \tag{18}$$

  - Limitations:
    * Length scale must be prescribed empirically
    * Cannot adapt to changing flow conditions
    * Limited accuracy for complex flows

- **Two-Equation Models (e.g., k-ϵ, k-ω):**
  - Solve transport equations for two turbulence quantities
  - For k-ϵ:

$$\nu_T = C_\mu \frac{k^2}{\epsilon} \tag{19}$$

    * ϵ equation provides length scale information
    * Automatic adaptation to flow conditions
  - For k-ω:

$$\nu_T = \frac{k}{\omega} \tag{20}$$

    * ω (specific dissipation rate) serves as inverse time scale
    * More numerically stable near walls
  - Advantages over single-equation models:
    * No need for prescribed length scales
    * Better representation of turbulence dynamics
    * Wider range of applicability

(20250214#118)

Explain about Spalart-Almaras model:

- Developed by Philippe Spalart and Steven Allmaras at Boeing for external aerodynamics applications:
  - Specifically designed for aircraft wing and body flows
  - Focused on attached and mildly separated external flows
  - Became popular in aerospace industry due to its robustness

- Single-equation model solving directly for eddy viscosity ν_T:

$$\frac{\overline{D}}{Dt}\nu_T = \nabla \cdot \left(\frac{\nu_T}{\sigma_\nu} \nabla \nu_T \right) + S_\nu(\nu, \nu_T, \Omega, |\nabla \nu_T|) \tag{21}$$

Model Term Analysis

- **Right-Hand Side Terms**:
  - First term - Anisotropic diffusion:

$$\nabla \cdot \left(\frac{\nu_T}{\sigma_\nu} \nabla \nu_T \right) \tag{22}$$

    - σ_ν is a model constant (typically $\approx 2/3$)
    - Allows eddy viscosity diffusion to differ from momentum diffusion
    - Better captures turbulent transport physics
  - Second term - Comprehensive source term:

$$S_\nu(\nu, \nu_T, \Omega, |\nabla \nu_T|) \tag{23}$$

    - Combines all physical effects influencing ν_T
    - Typically includes:
      - Production proportional to vorticity Ω
      - Destruction term depending on ν_T and wall distance
      - Trip terms for transition control
      - Non-linear dependencies on $|\nabla \nu_T|$

Boundary Conditions and Applications

- Boundary condition implementation:
  - Inflow: Specify ν_T based on turbulence intensity
  - Far-field: $\nu_T \to 0$
  - Walls: $\nu_T = 0$ with proper near-wall treatment
- Demonstrated success in key applications:
  - Transonic flows with shock-boundary layer interaction
  - Airfoil flows with mild separation
  - High-Reynolds number external aerodynamics
- Extension to hybrid RANS-LES approaches:
  - Detached Eddy Simulation (DES) methodology:
    - RANS mode in attached boundary layers
    - LES mode in separated regions and wakes
    - Automatic switching based on grid spacing
  - Particularly effective for:
    - Shock-induced separation
    - Unsteady wake dynamics
    - Buffet prediction

Model Advantages

- Computational efficiency (single transport equation)
- Robustness in adverse pressure gradients
- Natural transition to LES in hybrid methods
- Well-tuned empirical constants for aerospace applications

(20250221#119)

How are eddies defined?

- In turbulence theory, we consider flow structures (or eddies) at various scales. At the large scale, the flow has characteristic scales:
  - Length scale: L
  - Velocity scale: U_∞
- An **eddy** is a flow structure or region of fluid of size l that is:
  - *Organized in space*, meaning the flow within the eddy is coherent and exhibits a certain structure.
  - *Temporally connected*, meaning different parts of the eddy remain correlated over a certain timespan.
  - Characterized by:
    - ∗ A length scale l
    - ∗ A velocity scale $u(l)$
    - ∗ A timescale $\tau(l)$ (typically $\tau(l) \sim \frac{l}{u(l)}$)

(20250221#120)

On the basis of how Reynold's number is defined, argue how the effect of viscosity in the flow domain changes with increase in Reynold's number:

- The kinematic viscosity ν of a fluid is inversely proportional to the Reynolds number (Re), i.e.,

  $$\nu \propto \frac{1}{Re}.$$

  This relation reflects the fact that the Reynolds number is defined as:

  $$Re = \frac{UL}{\nu},$$

  where U is a characteristic velocity scale, and L is a characteristic length scale of the flow.

- Therefore, as the Reynolds number increases (i.e., when inertial forces dominate), the effect of viscosity diminishes. That is,

  > *As the Reynolds number increases, the viscous terms in the Navier-Stokes equations decay rapidly.*

  This means that for high Reynolds number flows (e.g., turbulence), viscous effects are confined to small regions (like boundary layers or small-scale eddies).

(20250221#121)

Obtain a form of integral length scale in terms of turbulent kinetic energy:

- Let us define the outer (or integral) scale of turbulence:

$$\ell_0 \sim \mathcal{L},$$

  where $\mathcal{L}$ is the characteristic length scale of the largest energy-containing eddies in the turbulent flow. These eddies are responsible for most of the turbulent kinetic energy.

- The corresponding velocity scale at this length is:

$$u_0 = u(\ell_0) \sim u_{\mathrm{rms}} = \left(\frac{2}{3}k\right)^{1/2},$$

  where k is the turbulent kinetic energy per unit mass and u_{rms} is the root mean square of the velocity fluctuations.

- The Reynolds number at the outer scale is:

$$Re_0 = \frac{u_0 \ell_0}{\nu} \gg 1,$$

  indicating that inertial forces dominate over viscous effects at large scales. This large Reynolds number is a hallmark of turbulent flow.

- **Generation of small scales:** In turbulence, small-scale motions (smaller eddies) are generated by non-linear internal interactions among larger eddies. These processes are often referred to as the turbulent energy cascade. The nonlinear terms in the Navier-Stokes equations are responsible for transferring energy from large to small scales.

- **Isotropy at small scales:** Although large-scale turbulence is generally anisotropic and dependent on boundary conditions and external forcing, the small-scale motions (in the inertial and dissipation ranges) tend to become:

  - *Statistically isotropic*, meaning their statistical properties are the same in all directions,
  - *Insensitive to external forcing and boundary shapes*, since these effects are filtered out through the cascade and the small-scale eddies are governed by local interactions.

  This is a key assumption in Kolmogorov's theory of turbulence.

(20250221#122)

State and explain Kolmogorov's first similarity hypothesis:

- **Hypothesis:** At sufficiently small scales of turbulent motion, the statistics of the velocity field are *isotropic* and *universal*, i.e., independent of the details of the large-scale flow and boundary conditions.
- **Motivation:**
  - Let ℓ_0 be the integral scale, i.e., the size of the largest energy-containing eddies.
  - Consider small-scale eddies of size $\ell \ll \ell_0$. These small eddies are the result of the energy cascade from larger eddies via nonlinear interactions.
  - At these small scales, it is assumed that the influence of the large-scale anisotropy has vanished due to the local and homogeneous nature of the interactions.
  - As a result, the velocity field statistics at small scales depend only on the local properties of the fluid and the energy dissipation rate.
- **Key Assumption:** The statistics at small scales depend only on:
  (i) the kinematic viscosity ν of the fluid (a property of the medium),
  (ii) the mean energy dissipation rate per unit mass ϵ (which characterizes the rate at which turbulent energy is transferred and dissipated).
- **Justification for dependence on ϵ and ν:**
  - The turbulent kinetic energy injected into the system (e.g., by a large-scale stirring mechanism) cascades down to smaller and smaller eddies until it is dissipated by viscous forces.
  - The energy dissipation mechanism acts dominantly at small scales and is governed by viscosity.
  - The dissipation rate ϵ is essential because it tells us how much energy per unit mass is being dissipated per unit time. Without ϵ, we cannot characterize the rate at which the turbulent motions are decaying.
  - The viscosity ν governs how strongly the fluid resists deformation, thus controlling the strength of dissipation at the smallest scales.
- **Conclusion:** Therefore, in the limit $\ell \ll \ell_0$, the turbulent motion is:

  Statistically isotropic and universal, governed only by ϵ and ν.

  This forms the basis for defining the Kolmogorov microscales using dimensional analysis.

(20250221#123)

- **Universal Equilibrium Range:**

- The range of scales where turbulent motions are independent of large-scale forcing or boundary conditions and are statistically isotropic.
- This range is referred to as the *universal equilibrium range.*
- It is characterized by scales ℓ much smaller than the integral scale $\mathcal{L}$:

$$\ell \ll \mathcal{L}.$$

More precisely, following Pope's convention, the condition is written as:

$$\ell \ll \ell_{\text{EI}},$$

where ℓ_{EI} is the scale at which the turbulence becomes effectively isotropic (EI = equilibrium isotropy).

- **Dimensional Analysis:** In the universal equilibrium range, the only relevant parameters governing small-scale turbulence are:
  - the energy dissipation rate per unit mass, ϵ,
  - the kinematic viscosity, ν.
- **Units:**

$$[\epsilon] = \left[\frac{u^3}{L}\right] = \left[\frac{(\text{length/time})^3}{\text{length}}\right] = \left[\frac{L^2}{T^3}\right],$$
$$[\nu] = [u][L] = \left[\frac{L}{T}\right][L] = \left[\frac{L^2}{T}\right].$$

- **Constructing a Velocity Scale from ϵ and ν:**

We seek a velocity scale u_η that depends only on ϵ and ν. Multiply their dimensions:

$$[\epsilon][\nu] = \left[\frac{L^2}{T^3}\right]\left[\frac{L^2}{T}\right] = \left[\frac{L^4}{T^4}\right] = [u]^4.$$

Taking the fourth root:

$$[u] = ([\epsilon][\nu])^{1/4}.$$

- **Kolmogorov Velocity Scale:**

Therefore, the velocity associated with the smallest scales of turbulence is:

$$u_\eta = (\epsilon\nu)^{1/4}.$$

- **Interpretation:**
  - The Kolmogorov velocity scale u_η characterizes the magnitude of velocity fluctuations at the smallest dynamically significant scales of the flow.
  - It emerges from the requirement that, in the universal equilibrium range, statistical properties depend only on ϵ and ν.
  - This velocity scale is a fundamental result of dimensional analysis under Kolmogorov's first similarity hypothesis.

(20250226#124)

Why does range of scales in turbulent flow increase with Reynolds number?

Reynolds number effect:

$$Re \uparrow \quad \Rightarrow \quad \text{Scale range } (l_0/\eta) \uparrow$$

This can be explained by the scale separation

$$\frac{l_0}{\eta} \sim Re^{3/4}$$

For fixed l_0, as Re increases, η decreases, and for fixed η, as Re increases, l_0 increases, thus widening the range of length scales containing turbulent kinetic energy.

(20250226#125)

Give the expression for energy spectrum in the inertial subrange

Energy spectrum in inertial subrange:

$$E(k) = C_k \epsilon^{2/3} k^{-5/3} \quad (C_k \approx 1.5)$$

(20250226#126)

What does Kolmogorov's second hypothesis say?

Kolmogorov's 2nd hypothesis (inertial subrange):

$$l_0 \gg l \gg \eta \quad \text{where:}$$

$$l_0 = \text{Integral scale (energy-containing)}$$
$$\eta = \text{Dissipation scale} = \left(\frac{\nu^3}{\epsilon}\right)^{1/4}$$

There is a state l intermediate range which depends only on ϵ, the dissipation rate. In this scale, the major dynamics is associated with the transfer of energy from the largest to the smallest scales. The universality in this scale arises from the dependence only on the ϵ. The viscous effects are negligibly small in these scales.

(20250226#127)

Formula for Kolmogorov time scale:

$$\begin{aligned} \epsilon &\sim \frac{u_0^3}{l_0} \quad \text{(energy cascade)} \\ \tau_\eta &= \left(\frac{\nu}{\epsilon}\right)^{1/2} \quad \text{(Kolmogorov time scale)} \end{aligned}$$

(20250226#128)

Obtain the velocity and length scales in intermediate scales. Obtain them in terms of integral and Kolmogorov scales as well.

All the scales in the intermediate range can be expressed through ϵ and l,

$$\begin{aligned} u(l) &= (\epsilon l)^{1/3} \\ \tau(l) &= (l^2/\epsilon)^{1/3} \\ \epsilon &= u^3/l = u_0^3/l \\ \implies u(l) &= u_0 \left(\frac{l}{l_0}\right)^{1/3} \\ \tau(l) &= \tau_0 \left(\frac{l}{l_0}\right)^{2/3} \end{aligned}$$

With

$$\epsilon = \nu \left(\frac{u_\eta}{\eta}\right)^2$$

which comes from $Re_\eta = u_\eta \eta/\nu = 1$, we have

$$u(l) = u_\eta \left(\frac{l}{\eta}\right)^{1/3}$$

$$\tau(l) = \tau_\eta \left(\frac{l}{\eta}\right)^{2/3}$$

(20250226#129)

What mechanisms control the changes in k, ϵ and λ?

- Turbulent kinetic energy (k) evolves through **production** (from mean shear), **dissipation** (ϵ), and **diffusion** (transport by turbulence/pressure).
- Dissipation rate (ϵ) changes via **production** (from vortex stretching), **dissipation** (viscous destruction), and **diffusion** (spatial transport).
- Taylor microscale (λ) adjusts through **diffusion** (scale interactions) and **dissipation** (affected by ϵ and ν), but has no direct production term.

$$\begin{aligned}
\frac{Dk}{Dt} &= \mathcal{P}_k - \epsilon + \mathcal{D}_k \\
\frac{D\epsilon}{Dt} &= C_{\epsilon 1}\frac{\epsilon}{k}\mathcal{P}_k - C_{\epsilon 2}\frac{\epsilon^2}{k} + \mathcal{D}_\epsilon \\
\lambda &= \sqrt{\frac{15\nu k}{\epsilon}}
\end{aligned}$$

where:

- $\mathcal{P}_k$: Production of k from mean shear
- $\mathcal{D}_k, \mathcal{D}_\epsilon$: Diffusion terms
- $C_{\epsilon 1}, C_{\epsilon 2}$: Model constants

(20250226#130)

What is the motivation behind the definition of integral length scales?

When correlations exist between velocity fluctuations which give rise to a non-trivial Reynold's stress, we can think of a length scale over which this correlation exists. This length scale corresponds to the integral length scale.

(20250226#131)

Define integral length scale:

The **integral length scale** (L) is defined as:

$$L \equiv \frac{1}{2E} \int_0^\infty \frac{E(k)}{k}\, dk$$

where:

- $E(k)$ is the turbulent kinetic energy spectrum
- $E = \int_0^\infty E(k)\, dk$ is the total turbulent kinetic energy
- k is the wavenumber ($k = 2\pi/\lambda$)

Alternative definition using two-point correlation:

$$L = \int_0^\infty f(r)\, dr$$

where $f(r)$ is the longitudinal velocity correlation function.

(20250226#132)

Give expressions for two-point correlation and auto-correlation functions:

Two-point correlation functions

Velocity correlation tensor:

$$R_{ij}(\mathbf{r}, \mathbf{x}, t) \equiv \langle u_i(\mathbf{x}, t) u_j(\mathbf{x} + \mathbf{r}, t) \rangle$$

where:

- u_i, u_j are velocity components
- $\mathbf{r}$ is separation vector
- $\langle \cdot \rangle$ denotes ensemble average

Longitudinal Auto-Correlation

$$f(r) \equiv \frac{\langle u_L(\mathbf{x}) u_L(\mathbf{x} + r\mathbf{e}_L) \rangle}{\langle u_L^2 \rangle}$$

where u_L is velocity component parallel to $\mathbf{r}$.

Transverse Auto-Correlation

$$g(r) \equiv \frac{\langle u_N(\mathbf{x}) u_N(\mathbf{x} + r\mathbf{e}_L)\rangle}{\langle u_N^2\rangle}$$

where u_N is velocity component normal to $\mathbf{r}$.

(20250303#133)

What happens at the jet periphery where it interfaces with the ambient fluid?

At the edges (periphery) of a jet, velocity gradients create instabilities. As a result, coherent structures form. These are organized, repeating flow patterns (like vortices or eddies) that form due to shear. Over time/distance, these structures break down into smaller, chaotic motions (turbulence).

(20250303#134)

For a stationary flow, what is the correlation between velocity fluctuations at $r = 0$ called?

For correlation, ensemble average is used if the statistical flow properties are time dependent.

$$R_{ij}(\boldsymbol{r}, t) = \langle u_i'(\boldsymbol{x}, t)u_j'(\boldsymbol{x} + \boldsymbol{r}, t)\rangle$$

If no time dependence, use time average,

$$R_{ij}\boldsymbol{r}, 0 = \langle u_i'(\boldsymbol{x})u_j'(\boldsymbol{x} + \boldsymbol{r})\rangle$$

When $r = 0$, R_{ij} becomes Reynold's stress. This quantity doesn't necessarily vanish in a turbulent flow.

$$Re^{\tau} = \langle u_i'u_j'\rangle$$

(20250303#135)

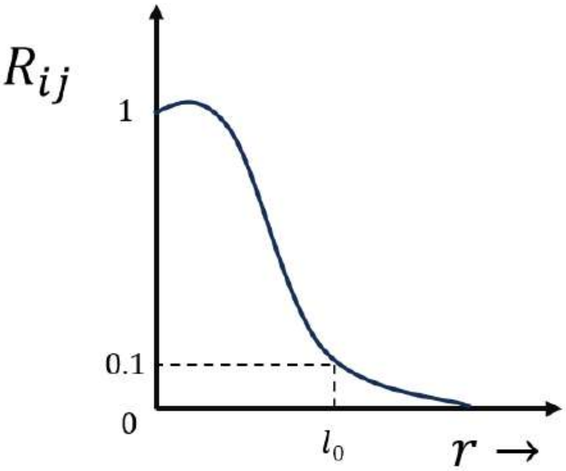
In simple words, explain integral length scale in terms of correlations

---

Velocity fluctuation signals are correlated over some distance. The length scale over which it is correlated corresponds to integral length scale.

(20250303#136)

Does correlations exist in isotropic flow (where there are no mean gradients)?



Correlations still exist but depend only on distance, not direction, due to rotational symmetry. In short, no net directional bias, fluctuations cancel out on average, but statistical relationships persist based on separation distance.

(20250303#137)

Longitudinal autocorrelation function expression:

The longitudinal autocorrelation function $R_{11}(r)$ for the fluctuating velocity component u_1' is defined as:

$$R_{11}(r) = \frac{\langle u_1'(\mathbf{x}, t)\, u_1'(\mathbf{x} + r\mathbf{e}_1, t)\rangle}{\langle u_1'^2\rangle} = f(\boldsymbol{r})$$

where:

- r is the separation distance along the x_1-direction,
- $\mathbf{e}_1$ is the unit vector in the x_1-direction,
- $\langle \cdot \rangle$ denotes ensemble or time averaging,
- $\langle u_1'^2 \rangle$ is the variance of u_1'.

For isotropic turbulence, $R_{11}(r)$ depends only on r (not direction) and satisfies:

$$R_{11}(0) = 1 \quad \text{(normalization)}, \quad R_{11}(-r) = R_{11}(r) \quad \text{(symmetry)}.$$

(20250303#138)

Expression for integral length scale in terms of longitudinal autocorrelation function:

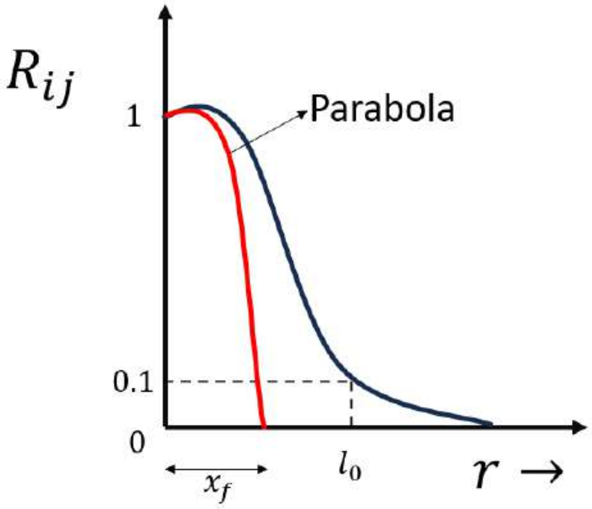
$$L_{11} = \int_0^\infty f(\boldsymbol{r})dr$$

$f(0) = 1,\ f'(0) = 0$ → because starts out flat.

(20250303#139)

How is Taylor microscale obtained?

Taylor microscale λ_f is obtained by fitting a parabola on the correlation function and taking the intercept of that parabola with the x-axis. The parabola that originates at $x = 0,\ y = 1$ has the same slope (tangent) there → osculating parabola. This function coincides with $f(\boldsymbol{r})$, slope is same, same tangent and also the curvature has to match which makes it an osculating parabola.



(20250303#140)

What's an osculating parabola?

An **osculating parabola** is the parabola that best approximates a given curve at a point by matching:

- Position ($\mathbf{r}$)
- Tangent direction ($\mathbf{T}$)
- Curvature (κ)

Definition: For a curve $y = f(x)$, the osculating parabola at $x = a$ has the form:

$$y = f(a) + f'(a)(x-a) + \frac{f''(a)}{2}(x-a)^2$$

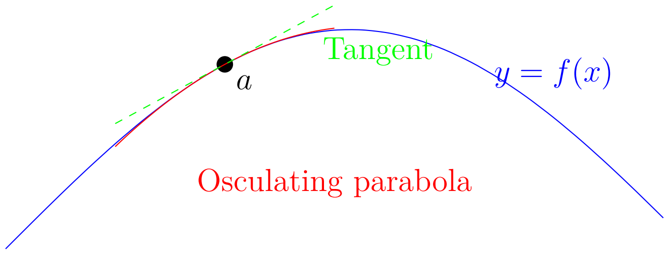
This is the *second-order Taylor approximation* of f at $x = a$.

Properties:

1. **Contact of order 2**: Matches f, f', and f'' at a.
2. **Curvature**: Shares the same curvature κ as f at a:

$$\kappa = \frac{|f''(a)|}{(1 + f'(a)^2)^{3/2}}$$

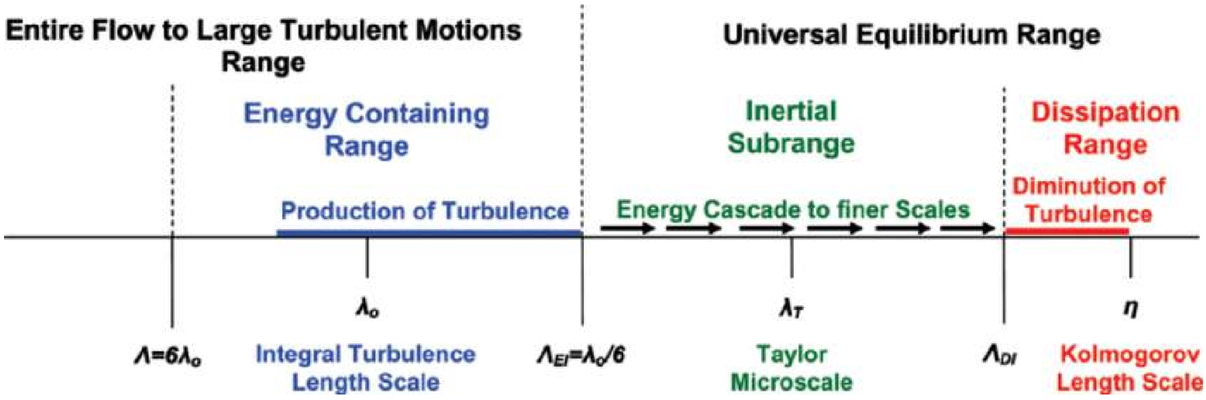
Visualization:



(20250303#141)

What is the relative value that Taylor microscale assumes compared to other scales? Where is Taylor microscale used?

Slightly larger than Kolmogorov scale (larger than few 10s of η), but lesser than integral scale. Certain turbulent flow properties scale with the Taylor microscale λ, making it a useful characterization parameter since λ can be inferred from integral-scale statistics. The Taylor-scale Reynolds number Re_λ is commonly used in literature to characterize certain turbulent flows. Near turbulent/non-turbulent interfaces (TNTIs), it represents the **scale of interaction** between turbulent and non-turbulent regions. For example:



- In **turbulent boundary layers** interacting with
- **non-turbulent outer flows**,

Taylor microscales λ become particularly relevant for flow description when significant mean velocity gradients $\nabla \overline{U}$ are present.

(20250303#142)

Write the expression for Taylor microscale in terms of longitudinal autocorrelation function:

$$\lambda_f = \left(-\frac{1}{2}f''(0)\right)^{-1/2}$$

(20250305#143)

What is the relationship between correlation and longitudinal autocorrelation function and transverse autocorrelation function:

Longitudinal and transverse autocorrelation functions are given by:

$$\begin{aligned} R_{11}(r) &= f(\boldsymbol{x})\langle u_1'^2\rangle = \langle u_1'(\mathbf{x})u_1'(\mathbf{x}+r\mathbf{e}_1)\rangle \quad \text{(Longitudinal)} \\ R_{22}(r) &= g(\boldsymbol{x})\langle u_2'^2\rangle = \langle u_2'(\mathbf{x})u_2'(\mathbf{x}+r\mathbf{e}_1)\rangle \quad \text{(Transverse)} \end{aligned}$$

Key properties for HIT case:

- At $r = 0$: $R_{11}(0) = R_{22}(0) = \frac{2}{3}k$ (where k is turbulent kinetic energy)
- Transverse correlations decay faster at large r

(20250305#144)

State Taylor's frozen flow hypothesis. How can I use it to relate temporal and spatial autocorrelation?

Taylor's Frozen Flow Hypothesis: Under the assumption that turbulent eddies advect past measurement probes *without significant distortion*, we can relate temporal measurements to spatial structure:

$$\frac{\partial}{\partial t} \approx -U\frac{\partial}{\partial x}$$

where:

- U is the **local mean velocity**
- The approximation holds when eddy evolution is slow compared to advection ($\tau_{\text{eddy}} \gg L/U$)

Probe Measurement
$\Downarrow$
Temporal autocorrelation $R(\tau)$
$\Downarrow$
Spatial correlation $R(r)$ via $r = U\tau$

The figure shows the conversion of time-domain measurements to spatial correlations using the frozen flow hypothesis.

Key implications:

- Valid when turbulent intensity $u'/U \ll 1$
- Enables estimation of integral scales from single-point measurements

- Widely used in wind tunnel experiments and atmospheric studies

(20250305#145)

Give formulae for longitudinal and transverse integral length scales. How are they related in isotropic turbulence?

Longitudinal integral length scale:

$$\int_0^\infty f(\boldsymbol{r})d\boldsymbol{r} = L_{11}$$

Transverse integral length scale:

$$\int_0^\infty g(\boldsymbol{r})d\boldsymbol{r} = L_{22}$$

In isotropic turbulence,

$$L_{22} = \frac{1}{2}L_{11}$$

(20250305#146)

Obtain Taylor microscales based on longitudinal and transverse velocity correlation functionos.

The Taylor microscale λ can be determined from both longitudinal and transverse velocity correlation functions:

For the longitudinal correlation function $R_{11}(r)$:

$$\lambda_f^2 = -\frac{2}{\left.\frac{d^2 f(\boldsymbol{r})}{dr^2}\right|_{r=0}} \tag{24}$$

For the transverse correlation function $R_{22}(r)$:

$$\lambda_g^2 = -\frac{2}{\left.\frac{d^2 g(\boldsymbol{r})}{dr^2}\right|_{r=0}} \tag{25}$$

For discrete experimental data, the second derivative can be approximated:

$$\left.\frac{d^2 f}{dr^2}\right|_{r=0} \approx \frac{2[R_{11}(0) - R_{11}(\Delta r)]}{(\Delta r)^2} \tag{26}$$

$$\left.\frac{d^2 g}{dr^2}\right|_{r=0} \approx \frac{2[R_{22}(0) - R_{22}(\Delta r)]}{(\Delta r)^2} \tag{27}$$

where Δr is the smallest spatial separation in measurements.

For homogeneous isotropic turbulence,

$$\lambda_g = \frac{\lambda_f}{\sqrt{2}}$$

(20250305#147)

What is the formula for ϵ in terms of λ_g?

From symmetry and using $\epsilon = 2\nu S_{ij} S_{ij}$, we can show that

$$\epsilon = \frac{15\nu u'^2}{\lambda_g^2}$$

where u_1^2 → obtained from measurements and λ_g → obtained from correlation function.

(20250305#148)

Obtain a length scale L based on amplitude of fluctuations and dissipation rate. Use this L to obtain a Re_L. Compare λ_g with L.

If we take length scale as $L = k^{3/2}/\epsilon$ → measured based on amplitude of fluctuations and dissipation rate of those fluctuations, we have

$$Re_L = \frac{k^{1/2} L}{\nu} = \frac{k^2}{\epsilon \nu}$$

$$\frac{\lambda_g}{L} = \sqrt{10} Re_L^{-1/2}$$

We already know

$$\frac{\eta}{L} = Re^{-3/4}$$

using which we can say,

$$\lambda_g = \sqrt{10}\;\eta^{2/3}\;L^{1/3}$$

$\eta < \lambda_g < L$. η, λ_g → intrinsic scales based on turbulence

L → external geometric turbulence length scale, like diameter of jet, size of the largest eddy etc.

Note: Literature reported Re is often Taylor Reynolds number Re_λ.

(20250305#149)

What is the expression for energy spectrum function? Based on it, find turbulence kinetic energy for a turbulent flow.

Energy spectrum function $E(\kappa)$:

$$k_{\kappa_a,\kappa_b} = \int_{\kappa_a}^{\kappa_b} E(\kappa) d\kappa$$

$E(k)$ → density of energy present the wavenumber band $[\kappa; \kappa + d\kappa]$.

For a turbulent flow, the turbulence kinetic energy would be

$$k = \int_0^\infty E(\kappa) d\kappa$$

and dissipation rate would be

$$\epsilon = 2\nu \int_0^\infty \kappa^2 E(\kappa) d\kappa$$

Note:

- **Kinetic Energy** (k) depends on the *squares of velocities*:

$$k = \frac{1}{2}\langle u_i' u_i' \rangle = \frac{1}{2}\langle u'^2 + v'^2 + w'^2 \rangle \tag{28}$$

- **Dissipation Rate** (ε) depends on the *velocity derivatives*:

$$\varepsilon = 2\nu \langle s_{ij} s_{ij} \rangle = \nu \left\langle \left(\frac{\partial u_i'}{\partial x_j} + \frac{\partial u_j'}{\partial x_i} \right)^2 \right\rangle \tag{29}$$

  where ν is kinematic viscosity and s_{ij} is the strain rate tensor.

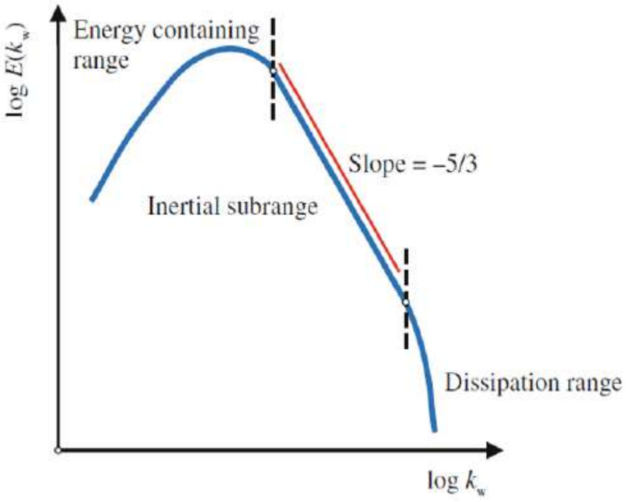
If u is a function with Fourier transform $\hat{u}$, Fourier transform of du/dt will be $i\kappa\hat{u}$.

(20250305#150)

Derive $E(\kappa)$ relation with ϵ and κ in the inertial subrange:

Starting from Kolmogorov's hypotheses for homogeneous isotropic turbulence:

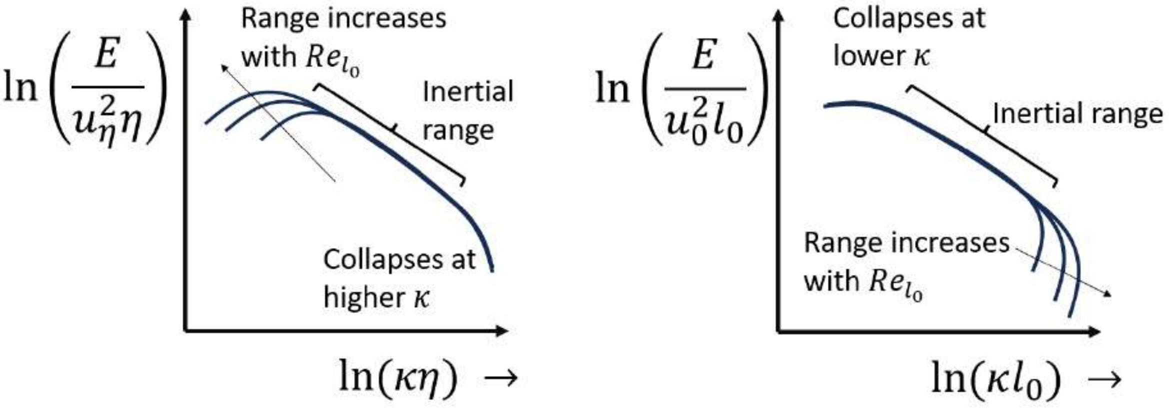
1. **Assumption**: Energy cascade depends only on:
   - Wavenumber κ
   - Dissipation rate ϵ
2. **Dimensional analysis**:
$$\begin{aligned}[E(\kappa)] &= \mathrm{m}^3/\mathrm{s}^2 \\ [\epsilon] &= \mathrm{m}^2/\mathrm{s}^3 \\ [\kappa] &= \mathrm{m}^{-1}\end{aligned}$$
3. **Power law ansatz**:
$$E(\kappa) = C\epsilon^a\kappa^b \tag{30}$$
4. **Dimensional consistency requires**:
$$\begin{aligned}a &= 2/3 \\ b &= -5/3\end{aligned}$$
5. **Final form**:
$$E(\kappa) = C_K\epsilon^{2/3}\kappa^{-5/3} \tag{31}$$
   where $C_K \approx 1.5$ is the Kolmogorov constant.



(20250305#151)

How does energy spectrum scaled with Kolmogorov length scale and integral length scale look like?

We've used Kolmogorov length scale η and integral length scale l_0 to obtain appropriate scaling for energy and wavenumber here.


$\ln\left(\frac{E}{u_\eta^2\eta}\right)$
Range increases with Re_{l_0}
Inertial range
Collapses at higher κ
$\ln(\kappa\eta)$ →
$\ln\left(\frac{E}{u_0^2 l_0}\right)$
Collapses at lower κ
Inertial range
Range increases with Re_{l_0}
$\ln(\kappa l_0)$ →

(20250307#152)

What simplification can be done in forcing terms of linear PDE as opposed to a non-linaer PDE?

For **linear PDEs**, the solution behavior allows decomposition of forcing terms due to the *superposition principle*:

- **Decomposition**: Any forcing $f(x,t)$ can be expressed as a sum of impulse responses:
$$f(x,t) = \sum_i f_i(x,t)$$
- **Individual Solutions**: Each impulse f_i generates its own solution u_i:
$$\mathcal{L}u_i = f_i \quad \text{(for linear operator } \mathcal{L}\text{)}$$
- **Complete Solution**: The total solution is the sum:
$$u(x,t) = \sum_i u_i(x,t)$$

For **nonlinear PDEs** $\mathcal{N}(u) = f$:

- *Cross-terms appear*: $\mathcal{N}(u_1 + u_2) \neq \mathcal{N}(u_1) + \mathcal{N}(u_2)$
- *Interaction effects*: Solutions for individual impulses cannot be simply summed
- Requires alternative methods (perturbation, numerical schemes, etc.)

$$\text{Linear} \quad \mathcal{L}(u_1 + u_2) = \mathcal{L}u_1 + \mathcal{L}u_2 \quad \text{vs.} \quad \text{Nonlinear} \quad \mathcal{N}(u_1 + u_2) \neq \mathcal{N}u_1 + \mathcal{N}u_2$$

(20250307#153)

Why are localized large gradients computationally tractable, while global ones are not?

- **Localized sharp changes** are computationally manageable:
  - Can use adaptive mesh refinement
  - May employ shock-capturing schemes in affected regions
- **Globally prevalent sharp changes** create numerical challenges:
  - Require uniformly high resolution
  - Induce stiffness in the system
  - Limit stable time-step sizes (CFL condition)

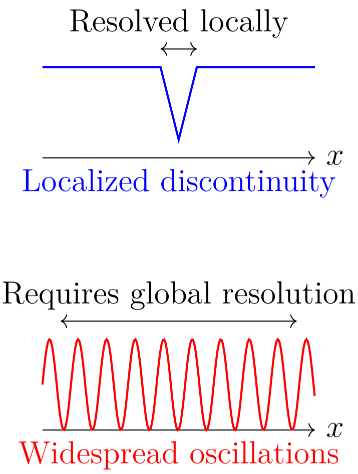


Figure 1: Contrast between localized vs. globally prevalent sharp spatial variations

Mathematical Interpretation

For a solution $u(x)$ with characteristic length scales:

$$\text{Difficulty} \propto \left(\frac{\text{Domain size } L}{\text{Smallest feature } \delta} \right)^d \times \text{Rejection rate}$$

where d is dimensionality. Widespread small δ values:

- Increase required degrees of freedom
- Create communication overhead in parallel computing
- May require implicit time-stepping globally

(20250307#154)

How does one get a rough estimate of the number of grid points required to simulate a turbulent flow?

The Kolmogorov scale η relates to the integral scale l_0 through the Reynolds number Re:

$$\frac{\eta}{l_0} \sim Re^{-3/4} \tag{32}$$

This implies the grid spacing Δx must resolve the smallest dynamically significant scales:

- **Grid spacing requirement**:

$$\frac{\Delta x}{L} \sim \frac{\eta}{l_0} \sim Re^{-3/4} \tag{33}$$

- **Three-dimensional resolution**:
$$\frac{\Delta x}{L}\frac{\Delta y}{L}\frac{\Delta z}{L} = \left(\frac{1}{N_x}\right)\left(\frac{1}{N_y}\right)\left(\frac{1}{N_z}\right) \sim Re^{-9/4} \tag{34}$$
- **Total grid points**:
$$N^3 \sim Re^{9/4} \tag{35}$$

Practical Implications

For typical high-Reynolds number flows ($Re \sim 10^4$ to 10^8):

$$N^3 \sim (10^4)^{9/4} = 10^9 \quad \text{(billions of points)}$$
$$N^3 \sim (10^8)^{9/4} = 10^{18} \quad \text{(exascale requirement)}$$

Reynolds Number	Grid Points Required
10^4 (Wind tunnel)	$\sim 1 \times 10^9$
10^6 (Car aerodynamics)	$\sim 1 \times 10^{13}$
10^8 (Atmospheric flows)	$\sim 1 \times 10^{18}$

This demonstrates why:

- DNS becomes prohibitively expensive at high Re
- LES/RANS modeling is necessary for practical engineering flows
- Current supercomputers can barely handle $Re \sim 10^6$ in DNS

(20250307#155)

How does time step constraint come into the picture of $Re^{9/4}$ issue?

For numerical stability in turbulent flow simulations, the time step Δt is constrained by both:

1. CFL Condition (Information Propagation)

The Courant-Friedrichs-Lewy condition requires:

$$\Delta t \leq C\frac{\Delta x}{u_{\max}} \sim \Delta x$$

where:

- C is the Courant number ($C \leq 1$ for stability)
- $u_{\max}$ is the maximum flow velocity
- Ensures information doesn't propagate faster than numerical scheme allows

2. Viscous Stability Constraint

For viscous flows, an additional requirement appears:

$$\Delta t \leq \frac{(\Delta x)^2}{2\nu} \sim (\Delta x)^2 \tag{36}$$

where ν is kinematic viscosity.

Combined Effect for High Reynolds Numbers

Given the spatial resolution requirement from the Kolmogorov scale:

$$\Delta x \sim L\, Re^{-3/4} \tag{37}$$

The time step becomes doubly constrained:

$$\Delta t \sim \Delta x \sim Re^{-3/4} \quad \text{(CFL condition)} \tag{38}$$

$$\Delta t \sim (\Delta x)^2 \sim Re^{-3/2} \quad \text{(Viscous constraint)} \tag{39}$$

Key Implications:

- Viscous constraint dominates at high Re
- Total time steps needed scale as $Re^{3/2}$
- Combined with spatial $Re^{9/4}$ requirement, DNS becomes:

$$\text{Total operations} \sim Re^{9/4} \times Re^{3/2} = Re^{15/4} \tag{40}$$

- For $Re = 10^6$, this leads to $\sim 1 \times 10^{11}$ operations per time unit

(20250307#156)

Which simulation method uses no model for turbulence? What main assumption is used for this method to simulate combustion? In early years of using this method, how was the flow treated to be made feasible to use in early computers?

Direct Numerical Simulation (DNS) was used for the numerical solution of Navier-Stokes equation without any model for turbulence. The model assumes infinitely fast chemistry to account for the time scale of combustion.

Computational Constraints

For many years, Direct Numerical Simulation (DNS) of practical turbulent flows was computationally infeasible due to:

- High Reynolds number requirements ($N^3 \sim Re^{9/4}$)
- Small time step constraints ($\Delta t \sim Re^{-3/2}$)
- Limited computational resources

Periodic Flow Simplification

To make simulations tractable, researchers employed:

$$\mathbf{u}(\mathbf{x}, t) = \mathbf{u}(\mathbf{x} + L\mathbf{e}_i, t) \quad \text{(Exact periodicity)} \tag{41}$$

Key characteristics:

- **Not just boundary conditions**: The entire flow field is strictly periodic
- **Mean gradient elimination**: $\boldsymbol{\nabla}\bar{\mathbf{u}} = 0$ (no shear)
- **Homogeneous turbulence**: Statistical properties invariant under translation

Physical Justification

This approach was reasonable for studying:

- Fundamental turbulence mechanisms
- Isotropic decay
- Small-scale universality (Kolmogorov's hypotheses)

Real Flow	**Periodic Approximation**
Mean gradients present	$\boldsymbol{\nabla}\bar{\mathbf{u}} = 0$
Inhomogeneous	Homogeneous
Complex boundaries	Infinite domain replication

Figure 2: Comparison between physical flows and periodic DNS approximation

(20250307#157)

Explain briefly about DNS of HIT:

Spectral Method Fundamentals

Early DNS focused on HIT in a periodic box, evolving Fourier modes $\hat{\mathbf{u}}(\boldsymbol{\kappa}, t)$ instead of physical fields $\mathbf{u}(\mathbf{x}, t)$:

$$\mathbf{u}(\mathbf{x}, t) = \sum_{\boldsymbol{\kappa}} \hat{\mathbf{u}}(\boldsymbol{\kappa}, t) e^{i\boldsymbol{\kappa}\cdot\mathbf{x}} \tag{42}$$

For a function sampled at N points:

- Yields N Fourier coefficients (complex)
- 0 to $N/2$ independent modes (Nyquist)
- Derivatives computed via FFT: $\mathcal{O}(N \log N)$ vs. finite difference $\mathcal{O}(N^2)$

Nonlinear Term Challenge

The Navier-Stokes nonlinear term becomes a convolution in Fourier space:

$$\mathcal{F}[(\mathbf{u} \cdot \nabla)\mathbf{u}] = i \sum_{\mathbf{p}+\mathbf{q}=\boldsymbol{\kappa}} (\boldsymbol{\kappa} \cdot \hat{\mathbf{u}}(\mathbf{p}))\hat{\mathbf{u}}(\mathbf{q}) \tag{43}$$

Solution approaches:

- **Pseudospectral method**:
  1. Transform to physical space ($\mathcal{O}(N \log N)$)
  2. Multiply locally ($\mathcal{O}(N)$)
  3. Transform back to Fourier space ($\mathcal{O}(N \log N)$)
- **Aliasing issue**: Handled via 3/2 rule (zero-padding)

Historical and Modern Context

	Early DNS	Modern Capability
Resolution	$N = 32^3$	$N = 4096^3$
Domain size	$\sim 6 - 7L$	$\sim 10L$
Turbulence type	Decaying	Forced stationary

Key Limitations

- **No mean gradients**: $\nabla\bar{\mathbf{u}} = 0$
- **Decaying turbulence**: Requires forcing for stationarity
- **Forcing method**: Amplitude adjustment of low-κ modes:

$$E(\kappa) = \begin{cases} \text{Modified} & \kappa \leq \kappa_f \\ \text{Natural} & \kappa > \kappa_f \end{cases} \tag{44}$$

(20250307#158)

What are some simple cases solved during the early days of DNS?

- Homogeneous and Isotrpoic Turbulence
- Uniform imposed shear
- Plane mixing layer
- Plane channel flow

(20250307#159)

Explain briefly about plane channel flow early days DNS:

Boundary Conditions

- **Periodic directions** (homogeneous):
  - Streamwise (x): $\mathbf{u}(x, y, z) = \mathbf{u}(x + L_x, y, z)$

  - Spanwise (z): $\mathbf{u}(x, y, z) = \mathbf{u}(x, y, z + L_z)$

- **Wall-normal direction** (y):
  - Non-periodic due to viscous boundary layers
  - No-slip conditions: $\mathbf{u}(x, 0, z) = \mathbf{u}(x, 2h, z) = 0$

Spectral Discretization

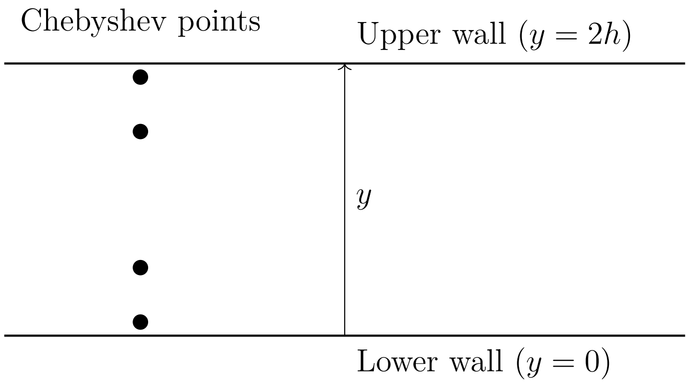


Figure 3: Plane channel flow configuration with spectral discretization

Chebyshev-Fourier Method

For wall-normal direction ($y \in [-1, 1]$):

$$T_n(y) = \cos\bigl(n \cos^{-1} y\bigr), \quad n = 0, 1, 2, \ldots, N \tag{45}$$

Transformation to Fourier space:

$$u(x, y, z) = \sum_{m=-M/2}^{M/2} \sum_{n=0}^{N} \sum_{p=-P/2}^{P/2} \hat{u}_{mnp} T_n(y) e^{i(\alpha_m x + \beta_p z)} \tag{46}$$

where $\alpha_m = 2\pi m/L_x$, $\beta_p = 2\pi p/L_z$.

Advantages of Spectral Methods

- **Exponential convergence** for smooth solutions
- **Exact differentiation** in Fourier space:

$$\frac{\partial}{\partial x} \to i\alpha_m \tag{47}$$

- **Superior accuracy** compared to finite differences:

$$\text{Error} \sim \begin{cases} e^{-cN} & \text{(Spectral)} \\ N^{-k} & \text{(Finite difference)} \end{cases} \tag{48}$$

Implementation Notes

- Chebyshev points clustered near walls:

$$y_j = \cos\left(\frac{j\pi}{N}\right), \quad j = 0, \ldots, N \tag{49}$$

- Nonlinear terms handled via pseudospectral approach
- Aliasing controlled with 3/2 padding rule

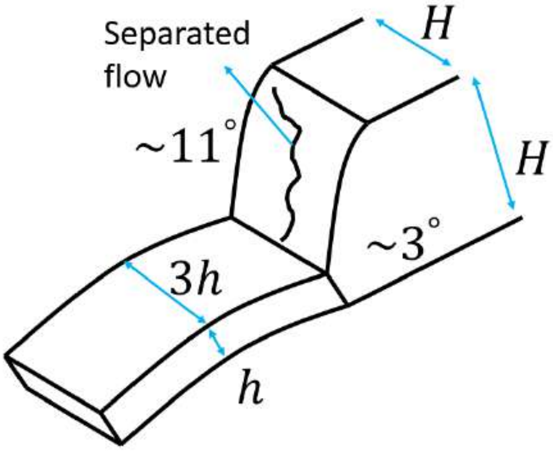
(20250310#160)

What can be said about the reliability of results from RANS?

Predicting turbulent flow with numerical models: RANS should work, but not necessarily. Compared to DNS which is too expensive, RANS is computationally feasible. It can give qualitatively right answers, and quantitatively right answers within some error bound.

RANS method however need not reliably capture some flow phenomena, such as flow separation. One RANS model may give separation, while another one need not. We don't need to have complex geometries to have this problem. In fact in complicated flows, it is less prone to occur, for example in complicated air passages of compressors and turbines, the air in secondary passages not really calculated in more detail in RANS.

Example scenario: We have a rectangular duct transitioning into a square duct. The flow separates off the top wall in experiment. Separation curve extends all the way to upper plane. In $k-\epsilon$, $k-\omega$ models, flow separates off the right wall only, and not the upper wall. If we were to look at heat transfer characteristics, attached region wouldu have different characteristics compared to detached region → RANS gives completely wrong result for the heat transfer characteristics.



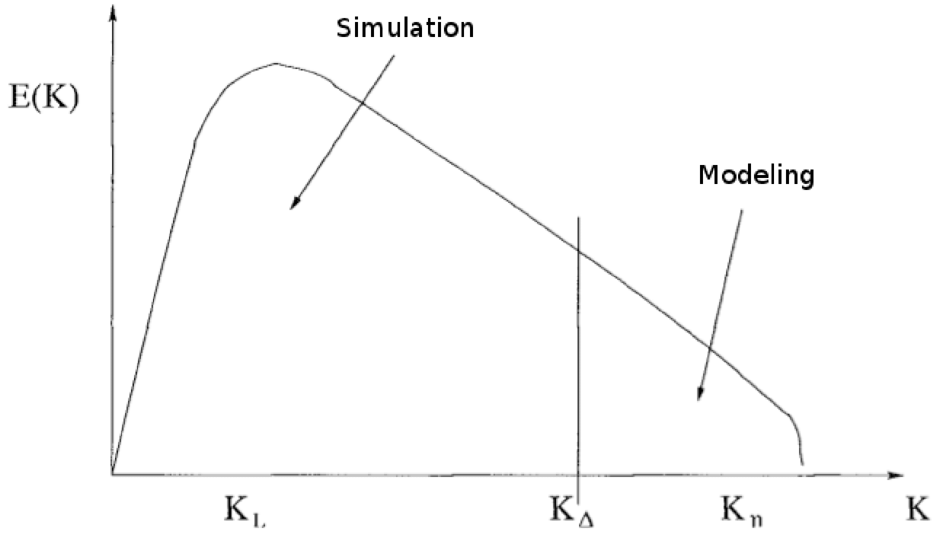
Moral of the story: Use RANS with discretion, even if its an elementary looking flow.

Note: In the same test case, LES was able to produce results matching with the experiment.

In practical flow devices like compressor cascade, we can see how using RANS might give us trouble. It is shown to perform well for flow near pressure surface, but flow near suction surface was not well captured by RANS.

(20250310#161)

Explain Large Eddy Simulation briefly:



Energy Cascade Dynamics

- **Net energy transfer**: Predominantly from large to small scales (forward cascade)

$$\Pi(\kappa_c) = \int_{\kappa_c}^{\infty} T(\kappa) d\kappa > 0$$

  where $T(\kappa)$ is the energy transfer function
- **Backscatter**: Local inverse transfer (small → large scales)
  - Caused by vortex stretching mechanisms
  - Typically $\sim 10-20\%$ of forward transfer
  - Modeled via stochastic terms in LES
- **Universal range**: For $\kappa \gg \kappa_c$ but $\kappa \ll \kappa_\eta$

$$E(\kappa) = C_K \epsilon^{2/3} \kappa^{-5/3} \tag{50}$$

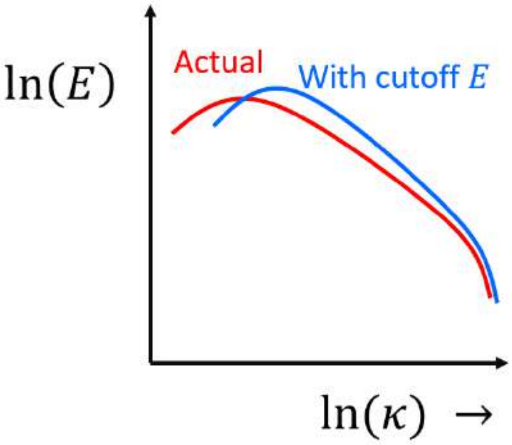
Scale Dependence

The relative importance of backscatter depends on:

$$\frac{\text{Backscatter}}{\text{Forward transfer}} \sim \frac{E_{sgs}}{E_{resolved}} \tag{51}$$

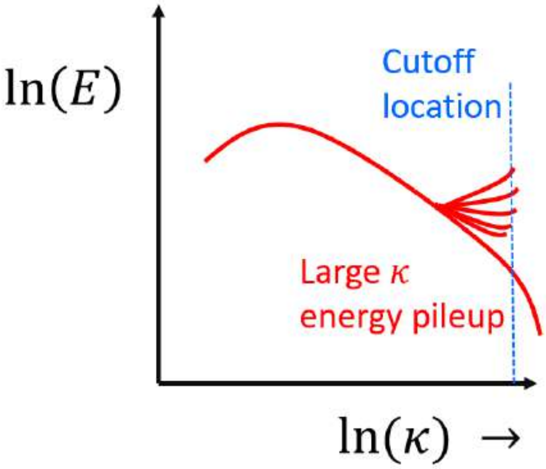
Regime	**Energy Ratio**	**Backscatter Effect**
Well-resolved LES	$E_{sgs}/E_{resolved} \ll 1$	Negligible
Coarse LES	$E_{sgs}/E_{resolved} \sim 0.1$	Significant

(20250310#162)
What is the effect of neglecting reverse energy transfer of small scales in LES?



If we neglect the effect of energy transfer of small scales, we end up with distorting the energy spectrum. A proper model has to be chosen to account for the backscatter. Thus the behavior of $E(\kappa)$ vs κ depends on the model of LES chosen.

(20250310#163)
What happens if we don't account for dissipation beyond the cutoff wavenumber region in LES?



Spectral Truncation Effects

When solving only for a subset of wavenumbers $\kappa \leq \kappa_c$ in LES:

$$\mathbf{u}(\mathbf{x},t) = \sum_{\|\boldsymbol{\kappa}\| \leq \kappa_c} \hat{\mathbf{u}}(\boldsymbol{\kappa},t) e^{i\boldsymbol{\kappa}\cdot\mathbf{x}} \tag{52}$$

We introduce two key approximations:

- **Neglected backscatter**: Energy transfer from $\kappa > \kappa_c$ to $\kappa \leq \kappa_c$
- **Effective dissipation**: Missing viscous effects at $\kappa > \kappa_c$

Model Dependence

The energy spectrum behavior depends critically on the subgrid-scale (SGS) model:

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u}\cdot\nabla)\mathbf{u} = -\nabla p + \nu\nabla^2\mathbf{u} + \underbrace{\nabla\cdot\boldsymbol{\tau}}_{\text{SGS term}} \tag{53}$$

We have to account for the dissipation in the neglected region in order to prevent the energy pileup at large wavenumbers κ.

(20250312#164)
What is the nature of dependence of eddy viscosity on grid spacing in Smagorinsky model for LES?

(20250312#165)
State general advantages and disadvantages of RANS and DNS and explain what happens when they try to simulate a separation bubble scenario in flow over a smooth surface:

(20250312#166)
Is there a transition sensitive modeling for RANS? Explain:

(20250312#167)
Explain with an example as to what the basis for large eddy simulation is:

(20250312#168)
What causes the development of smaller scales in high Reynolds number turbulent flows?

(20250312#169)
What are the mathematical and physical definitions of LES?

(20250312#170)
Obtain large scale part of a fluid with filtering:

(20250312#171)
Give the motivation for filtering to obtain large scale dynamics of a fluid flow:

(20250312#172)
What are some possible low pass filtering operations that could be used for obtaining large scale dynamics of turbulent flow?

(20250312#173)
For Gaussian filtering operation, what is the effect of using a narrower and broader filter?

(20250312#174)
Explain how convolution in physical space being product in spectral space helps with the filtering process:

(20250312#175)

For 1D convection problem, obtain dispersion relation and justify the application of low pass filter to weed out small scales.

(20250312#176)

Obtain LES equations for the problem

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0$$

(20250312#177)

Explain what happens if $f(u)$ is linear in u vs $f(u)$ is non-linear in u.

(20250312#178)

Explain modeling of subgrid scale stress for the obtained LES equation:

(20250312#179)

Higher the Re, better the subgrid scale model. Why?

(20250312#180)

How to get the equation for kinetic energy evolution using the LES equation?

(20250312#181)

Compare and contrast ν_T and ν_{LES}:

(20250317#182)

Obtain the LES equivalent of Navier-Stokes equations

$$\frac{\partial u_i}{\partial t} + \frac{\partial}{\partial x_j}(u_j u_i) = -\frac{1}{\rho}\frac{\partial P}{\partial x_i} + \nu \frac{\partial^2 u_i}{\partial x_j x_j}$$

Low pass filtering:

$$\overline{u_i} = G * u_i$$

Apply to Navier-Stokes equations,

$$\frac{\partial \overline{u_i}}{\partial t} + G * \frac{\partial}{\partial x_j}(u_j u_i) = -\frac{1}{\rho}\frac{\partial \overline{P}}{\partial x_i} + \nu \frac{\partial^2 \overline{u_i}}{\partial x_j x_j}$$

$$\frac{\partial \overline{u_i}}{\partial t} + \frac{\partial}{\partial x_j}(\overline{u_j}\ \overline{u_i}) + \left[G \frac{\partial u_j u_i}{\partial x_j} - \frac{\partial \overline{u_j}\ \overline{u_i}}{\partial x_j}\right] = -\frac{1}{\rho}\frac{\partial \overline{P}}{\partial x_i} + \nu \frac{\partial^2 \overline{u_i}}{\partial x_j x_j}$$

This is the equation in $F = ma$ form. Here,

$$G \frac{\partial u_j u_i}{\partial x_j} - \frac{\partial \overline{u_j}\ \overline{u_i}}{\partial x_j}$$

is the **subgrid scale stress tensor**. It can be thought of as an additional forcing term coming from the unresolved scales (actually subgrid scale contribution "acceleration" term).

(20250317#183)

What are some additional source terms which can be added to Navier-Stokes equations?

- **Gravitational force**
- **Electromagnetic forcing**: can be Lorentz force acting on a charged fluid (magnetohydrodynamics)
- **Temperature gradient induced forcing**: can introduce additional effects like buoyancy forces (in natural convection) or thermal stresses in viscous forces.

  In regions where temperature gradient is small or negligible, the viscosity of the fluid remains approximately uniform. This means we don't need additional terms to model viscosity.

When there is strong temperature gradient, viscosity can vary significantly. This is important particularly in high-temperature flows like in combustion, plasma physics, atmospheric re-entry etc. The change in viscosity due to temperature differences leads to additional stresses in the fluid.

(20250317#184)

How is the goal of LES different from RANS?

In LES, the numerical method must be highly accurate in resolving turbulent at various scales. LES requires high resolution grids, low numerical dissipation and accurate time-stepping to ensure that large eddies evolve naturally without artificial damping.

In RANS, we average out all the turbulent fluctuations and model the effects of turbulence using turbulence models like $k - \epsilon$, $k - \omega$, Spalart-Allmaras etc. Since RANS doesn't directly resolve any turbulence, the numerical method doesn't have to be precise. The accuracy of RANS models depend on turbulence modeling rather than on grid resolution or numerical schemes.

(20250317#185)

What does "filtering" mean in LES?

In LES, filtering means separating the flow-field into

- Large-scale (resolved) components → $\overline{u_j}$ (filtered velocity)
- Small-scale (unresolved) components → u'_j (subgrid-scale fluctuations)

This is done using a mathematical filter that removes small turbulent structures below a certain scale.

(20250317#186)

Do we actually apply a filter to the velocity field in LES?

No. We don't compute the unfiltered field u_j and then apply filter to get $\overline{u_j}$. Instead we directly solve for $\overline{u_j}$ in the governing equations.

(20250317#187)

If we don't explicitly filter, how do we get the filtered fields?

Since small scales are not explicitly resolved, their effects appear as unknown terms in the equations. These are called subgrid-scale (SGS) terms, which need to be modeled using an SGS model (e.g., Smagorinsky, WALE, or dynamic models).

(20250317#188)

If no filtering is done explicitly, what acts as the filter in LES?

The numerical grid itself acts as an implicit filter! The grid resolution determines what turbulence scales are captured in $\overline{u_j}$ and what is left unresolved (which is handled by SGS models).

(20250317#189)

Write down the LES equivalent of Navier-Stokes equation, with the τ_{ji}^{SGS} representing the subgrid-scale forcing term:

$$\frac{\partial \overline{u_i}}{\partial t} + \frac{\partial}{\partial x_j}\left(\overline{u_j}\;\overline{u_i}\right) + \left[G\frac{\partial u_j u_i}{\partial x_j} - \frac{\partial \overline{u_j}\;\overline{u_i}}{\partial x_j}\right] = -\frac{1}{\rho}\frac{\partial \overline{P}}{\partial x_i} + \nu\frac{\partial^2 \overline{u_i}}{\partial x_j x_j}$$

$$\begin{aligned}\frac{\partial \overline{u_i}}{\partial t} + \frac{\partial}{\partial x_j}\left(\overline{u_j}\;\overline{u_i}\right) &= -\frac{1}{\rho}\frac{\partial \overline{P}}{\partial x_i} + \nu\frac{\partial^2 \overline{u_i}}{\partial x_j x_j} - \left[G\frac{\partial u_j u_i}{\partial x_j} - \frac{\partial \overline{u_j}\;\overline{u_i}}{\partial x_j}\right] \\ &= -\frac{1}{\rho}\frac{\partial \overline{P}}{\partial x_i} + \nu\frac{\partial^2 \overline{u_i}}{\partial x_j x_j} - \frac{\partial}{\partial x_j}\left(\tau_{ji}^{SGS}\right)\end{aligned}$$

(20250317#190)

Is subgrid stress tensor symmetric?

Yes; It is symmetric.

$$\tau_{ji}^{SGS} = \overline{u_j u_i} - \overline{u_j}\;\overline{u_i} = \overline{u_i u_j} - \overline{u_i}\;\overline{u_j} = \tau_{ij}^{SGS}$$

$$\tau_{ji}^{SGS} = \overline{u_j u_i} - \overline{u_j}\;\overline{u_i} = \mathcal{R}(u_i, \overline{u_i})A = \mathcal{R}(\overline{u_i})$$

(20250319#191)

What are the two different categories of subgrid scale modelling?

- Functional modelling
- Structural modelling

(20250319#192)

Explain the two categories of subgrid scale modelling:

In functional approach, we are not interested in the flow structures in the subgrid scales. Rather we look at the effect of subgrid scales as a function of the known, resolved flow quantities - like estimation of energy transfer between the resolved and unresolved scales.

Examples:

- Smagorinsky model
- Dynamic Smagorinsky model

Structural models attempt to reconstruct the small-scale structures of the flow. They focus on the physical nature and topology of the small-scale structures rather than on their energetic effects.

Examples:

- Approximate Deconvolution Models (ADM)
- Velocity estimation models that reconstruct subgrid velocity fields

- Scale similarity models

Structural models tend to be more computationally expensive than functional models, but they can provide detailed information about the flow structure.

(20250319#193)

Explain a priori analysis of SGS models with an example

SGS models are compared with high-resolution reference data (typically obtained from DNS) before implementing them in a full LES. It is different from a posteriori analysis, which tests the model in a fully running LES.

Let's say I run DNS of a grid with 1000^3 grid points. We want to look at the effect of a specific SGS model of our choice used in LES of the reduced mesh of size 200^3. We obtain a high resolution DNS dataset at first. Then a low pass filter is applied on the data to simulate the effect of LES, removing small scales. The filtered fields represente what an LES would resolve, while the filtered-out scales represent the subgrid scale effects.

The true SGS stress tensor is directly computed from

$$\tau_{ij} = \overline{u_i u_j} - \overline{u_i}\ \overline{u_j}$$

The SGS model's predicted stress tensor is then evaluated and the two are compared. Common performance metrics include

- Correlation coefficient
- Error norms (L_1, L_2, L_∞ for assessing deviations
- Energy transfer behavior - to make sure dissipation and backscattering is captured properly
- Anisotropy representation

(20250319#194)

What happens in a posteriori analysis of SGS models? What do we compare it against?

(20250319#195)
Explain scale-similarity model

The key idea in scale-similarity model is the fundamental assumption that the statistical properties at the smallest resolved scales are structurally similar to the largest unresolved (subgrid) scales. This similarity arises because turbulence exhibits a continuous range of scales and the smallest resolved scales can provide a good approximation to the largest unresolved (subgrid) scales.

Subgrid scale stress tensor is given by

$$\tau_{ij} = \overline{u_i u_j} - \overline{u_i}\ \overline{u_j}$$

where $\overline{u_i}$ is the filtered velocity field, $\overline{u_i u_j}$ is the filtered product of velocities.

Scale similarity model approximates τ_{ij} by comparing the filtered field at two different filter levels: the grid filter and the test filter. The test filter has larger width $\hat{\Delta}$ than the grid filter Δ. The model can be expressed as

$$\tau_{ij} = C\ \ \left(\overline{\tilde{u}_i\tilde{u}_j} - \overline{\tilde{u}_i}\ \overline{\tilde{u}_j}\right)$$

The term within the brackets is denoted L_{ij}, which is the Leonard stress.

(20250319#196)
Why does the LES computation during a posteriori analysis of the scale-similarity model diverge?

(20250319#197)
Explain hybrid/mixed models in SGS modelling:

(20250319#198)
How does velocity estimation models work?

(20250319#199)

Describe sum and difference interaction between two flow structures of wavenumbers k_1 and k_2 and explain what happens when the wavenumbers differ by small and large magnitudes:

(20250319#200)

How to choose a proper cutoff wavenumber for the LES simulation?

(20250319#201)

Explain approximate deconvolution models (ADM):

(20250319#202)

Explain wall-damping effects in the boundary layers of a wall-bounded flow and how it can interfere with the SGS models:

(20250319#203)

Explain about velocity estimation models in Large Eddy Simulation (LES):

- In LES, the velocity field is spatially filtered to resolve only the large-scale motions. The effect of the unresolved subgrid scales is modeled via the subgrid-scale (SGS) stress tensor:

$$\tau_{ij}^{\text{SGS}} = \overline{u_i u_j} - \overline{u}_i \, \overline{u}_j$$

  - u_i: Instantaneous velocity components.
  - $\overline{u}_i$: Filtered (resolved) velocity components.
  - $\overline{u_i u_j}$: Filtered product of the velocity components.

- The SGS stress τ_{ij}^{SGS} accounts for the influence of the subgrid-scale motions (smaller than the filter width) on the resolved scales. However, we only have access to $\overline{u}_i$, not u_i itself.

- Since $\overline{u_i u_j}$ is not directly computable from resolved variables, we must model or estimate this term. One approximation idea is:

  $$u_i^* \approx u_i$$

  where u_i^* is some estimated or modeled form of the velocity field. This introduces a modeling step to recover information lost through filtering.

- **Grid-based estimation method:**
  - Simulate the velocity field on a coarse grid, say 200^3, where only large-scale structures are resolved.
  - Also simulate on a finer grid, say 400^3, which captures a wider portion of the energy spectrum (including more of the inertial and small-scale structures).
  - Use the simulation results on the 400^3 grid to construct or calibrate a model for the subgrid-scale stress that can be applied on the 200^3 grid.
  - This helps to estimate how much spectral content is unresolved on the coarse grid.

- **Assumption:** The velocity field structure obtained on the 400^3 grid serves as a proxy for what is happening on even finer grids like 1000^3, at least until the model or numerical resolution fails to capture essential dynamics.

- **Modeling Strategy:**
  - The 400^3 simulation may eventually fail to capture finer dynamics (e.g., due to numerical errors or insufficient resolution).
  - But as long as it provides physically meaningful results, its energy spectrum can be used to model or infer the expected spectrum on coarser grids such as 200^3.
  - This approach is especially useful when full-scale simulations (e.g., 1000^3) are computationally infeasible.

- **Conclusion:**
  - Estimating subgrid stresses in LES is inherently a modeling problem due to the unavailability of $u_i u_j$.
  - By comparing simulations on multiple grid resolutions, we can leverage partial knowledge to build models that infer the impact of unresolved scales on the resolved field.
  - Such models aim to recover lost spectral energy and maintain fidelity of turbulence statistics across different resolutions.

(20250321#204)

Explain approximate deconvolution models (ADM):

(20250321#205)

What would the deconvolution operator look like for Gaussian and a sharp cutoff filter?

(20250321#206)

Give a barebones implementation sketch of ADM in LES:

(20250321#207)

What happens when we apply $QG = I$ for the full range in LES?

(20250321#208)

What are the structural and functional aspects of ADM?

(20250321#209)

Give final remarks of ADM in terms of the choices to be made, effect of grid refinement and the significance of I as compared to Q and G:

(20250321#210)

What is modified wavenumber?

(20250321#211)

How does the error fall-off behave in finite-difference and spectral approaches?

(20250322#212)

How is the approximate deconvolution operator built?

- ADM is a subgrid-scale modeling technique that aims to approximate the unfiltered velocity field from the filtered one using repeated filtering operations. The filtered governing equation for a conserved quantity (e.g., velocity component $\overline{u}$) is written as:

  $$\frac{\partial \overline{u}}{\partial t} + G * \frac{\partial}{\partial x} f(Q * \overline{u}) = 0$$

  where:
  - $\overline{u}$: Filtered (resolved) velocity field.
  - G: Low-pass filter operator.
  - Q: Approximate inverse of the filter G.
  - $f(\cdot)$: Nonlinear flux term.

- The idea is to apply the operator $Q \approx G^{-1}$ to approximate the unfiltered field from $\overline{u}$. The ideal relation is:

  $$QG = I \quad \text{or} \quad \hat{Q}(\kappa)\hat{G}(\kappa) = 1$$

  for a range of wave numbers κ, in the Fourier domain.

- Since G^{-1} may not exist or may be ill-posed for all frequencies, we construct an approximation:

  $$Q \approx \sum_{m=0}^{M} (I - G)^m$$

  This expression is inspired by the geometric series:

  $$\frac{1}{1-x} = \sum_{m=0}^{\infty} x^m \quad \text{for } |x| < 1$$

  Here, $x \equiv (I - G)$ is an operator, and the series is truncated at order M to yield a practical approximation. This construction treats Q as a sequence of filtering operations.

- **Interpretation:** The approximate inverse filter Q acts to recover the high-frequency content lost in $\overline{u}$ by repeated applications of the residual operator $(I - G)$. Each application adds back more detail that was suppressed by the original filter G.

- **Key Advantage:** This approach does not require periodic boundary conditions. Even for non-periodic functions (e.g., flow through a channel, where the domain is bounded by walls), the spatial filtering operation G and its approximate inverse Q can be applied locally.

- For example, in a channel flow simulation:
  - The velocity field is not periodic in the wall-normal direction.
  - Yet, we can still apply the spatial filter G (e.g., compact or box filter) and construct the approximation Q by repeated filtering.
  - This yields an estimate of the unfiltered field even near the walls, without relying on global periodicity.

- **Conclusion:** ADM reconstructs subgrid scales by inverting the filtering process approximately. The model provides a physics-based method to infer the contribution of unresolved scales to resolved quantities in LES, especially in complex geometries and non-periodic domains.

(20250322#213)

Give a brief description of pade filters and explain about regularization:

- **Padé Filters:** These are rational approximations to differential operators or filter transfer functions. They provide higher-order accuracy while preserving favorable stability properties. In numerical simulations of turbulent flows, Padé filters are used to attenuate high-frequency components without significantly affecting the resolved scales.
- **Regularization:** Regularization refers to modifying a mathematical or physical model to remove or control singularities in approximate solutions. This is essential when the original (idealized) problem is well-posed and regular, but its numerical or analytical approximations may introduce instabilities or nonphysical singular behavior.
- **General Idea:**
  - Suppose the exact (physical) problem has regular solutions—i.e., the solution remains smooth and finite.
  - However, in attempting to approximate this system (e.g., through discretization or simplified models), we might unintentionally introduce singularities or instabilities.
  - **Regularization** is any technique that ensures the approximate or numerical model also avoids unphysical singularities.
- **Example: Vortex Sheet**
  - A vortex sheet represents a discontinuity in the tangential component of the velocity field—i.e., a jump in vorticity across an infinitesimal thickness.
  - Mathematically, the sheet itself is a spatial singularity: vorticity is concentrated along a lower-dimensional manifold (a line or surface).
  - If such a sheet is perturbed, the perturbation grows rapidly.
  - **Growth rate depends on the wavelength:** shorter wavelengths lead to faster growth.
  - Hence, the system becomes violently unstable in the high-frequency regime, leading to an ill-posed or numerically explosive problem.
  - A regularized version assumes the sheet has *finite thickness*:
    - Only a finite range of wavelengths are unstable.
    - As thickness $\delta \to 0$, the finite-thickness sheet converges to the vortex sheet.
  - **Regularization approach:** Introduce a small viscosity term to spread out the vorticity over a small but finite region. This stabilizes the sheet and suppresses unbounded high-frequency growth.

- **Wall Boundary Inconsistencies in Inviscid Flow**
  - Inviscid equations (e.g., Euler equations) do not naturally enforce the no-slip boundary condition at solid walls.
  - This creates a discrepancy: physical boundary layers form due to viscosity, but the inviscid model predicts slippage.
  - To reconcile this, one strategy is to introduce a **vortex sheet** at the wall:
    - The vortex sheet compensates for the velocity jump and mimics the presence of a viscous boundary layer.
    - This allows the inviscid model to maintain physical realism near walls while avoiding singular behavior.
- **Conclusion:**
  - Regularization is an essential concept in computational fluid dynamics to prevent numerical instability and ensure well-posedness.
  - Examples include viscosity-based smoothing, filtering operations (such as Padé filters), or reformulating singular objects like vortex sheets into smooth approximations.
  - These techniques are crucial when dealing with multiscale and turbulent flows where fine-scale instabilities can dominate the solution behavior.

(20250322#214)

How does regularization come into the picture of LES?

- The full Navier–Stokes equations (with viscosity) are well-posed and typically do not exhibit finite-time singularities in physical settings:

  $$\frac{\partial u}{\partial t} + u \cdot \nabla u = -\nabla p + \nu \nabla^2 u$$

  The viscosity term $\nu \nabla^2 u$ acts as a regularizing mechanism that smooths the velocity field, particularly at small scales.

- However, in practice, we often solve an approximate version of the problem, such as in Large Eddy Simulation (LES), where the velocity field is spatially filtered to resolve only large-scale motions:

  $$\text{LES:} \quad \overline{u} = G * u$$

  This introduces the need to model the influence of unresolved (subgrid-scale) motions on the resolved (filtered) field.

- **Subgrid-Scale (SGS) Modeling:** Acts as a form of regularization for LES, similar in spirit to how viscosity regularizes the Navier–Stokes equations. It prevents the build-up of energy at unresolved scales and stabilizes the solution.

- The LES equation with approximate deconvolution modeling (ADM) and a regularizing SGS term may take the form:

$$\frac{\partial \overline{u}}{\partial t} + G * \frac{\partial}{\partial x} f(Q * \overline{u}) = \frac{1}{X}(I - G)\overline{u}$$

  - $\overline{u}$: Filtered velocity.
  - G: Low-pass filter.
  - Q: Approximate inverse of G.
  - $f(\cdot)$: Nonlinear flux term.
  - $(I - G)\overline{u}$: Acts as a residual capturing high-frequency content lost in filtering.
  - $\frac{1}{X}$: A scaling parameter that controls the strength of regularization.

- **Decomposition Interpretation:**
  - The second term on the LHS, $G * \partial_x f(Q * \overline{u})$, represents a **soft decomposition**—a smooth, filtered approximation of nonlinear interactions.
  - The first term on the RHS, $(I - G)\overline{u}$, is a **hard decomposition**—explicitly estimating the contribution of high-frequency content that was removed by filtering.

- **Connection to Regularization:** This framework introduces an *ad hoc* term on the RHS to stabilize the LES system. It is analogous to the viscosity term in the original Navier–Stokes equations, serving to regularize the ill-posedness caused by truncation of subgrid modes.

(20250322#215)

Explain Tikhonov's regularization:

- Consider an ill-posed inverse problem where we want to find x from noisy data $y \in \mathbb{R}^m$, given a known operator or matrix $A \in \mathbb{R}^{m \times n}$, such that:

$$Ax \approx y$$

  If A is ill-conditioned or rank-deficient, small perturbations in y (e.g., due to measurement noise) can lead to large errors in the solution x.

- **Tikhonov's idea:** Instead of solving the least-squares problem

$$\min_x \|Ax - y\|_2^2,$$

  solve a regularized version that adds a penalty term:

$$\min_x \left(\|Ax - y\|_2^2 + \lambda \|Lx\|_2^2 \right)$$

  where:

  - $\lambda > 0$ is the regularization parameter that controls the trade-off between fidelity to the data and smoothness/penalty.
  - L is a regularization matrix (often $L = I$ or a derivative operator).

- **Interpretation:**
  - The penalty term $\|Lx\|_2^2$ discourages solutions with large norm or high variation.
  - This is equivalent to imposing a prior belief that the true solution is "smooth" or "small" in some sense.

- **Solution:** The minimizer x^* satisfies:

$$(A^T A + \lambda L^T L)x^* = A^T y$$

  which is a well-posed linear system even if A is ill-conditioned or not full-rank.

- **Special case:** When $L = I$, this is called *ridge regression* in statistics and machine learning.

- **Applications:**
  - Image reconstruction (e.g., deblurring)
  - Solving Fredholm integral equations
  - Parameter estimation in inverse problems
  - Regression problems with multicollinearity

(20250322#216)

Explain Time-Stepping and Filtering in ADM with Euler Forward Scheme:

- In an Approximate Deconvolution Model (ADM), we evolve the filtered velocity field $\overline{u}$ in time using a two-step process involving:
  - Reconstruction of the unfiltered (deconvoluted) field $u^* \approx u$, using an approximate inverse filter Q.
  - Filtering the updated field back through a low-pass filter G.

- Consider time evolution from step $n \to n+1$ using Euler Forward time integration:

$$\overline{u}^{n+1} = \overline{u}^n - \Delta t \cdot G * \frac{\partial}{\partial x} f(Q * \overline{u}^n)$$

  where:
  - Δt: time step size.
  - $f(\cdot)$: nonlinear flux function.
  - Q: approximate inverse filter (used to reconstruct u^*).
  - G: low-pass spatial filter.

- Define the deconvoluted field:
$$u^{*n} = Q * \overline{u}^n$$
so the update becomes:
$$\overline{u}^{n+1} = G * u^{*n+1}$$
where u^{*n+1} is computed via an Euler step:
$$u^{*n+1} = u^{*n} - \Delta t \cdot \frac{\partial}{\partial x} f(u^{*n})$$

- Thus, the two-step time-stepping process is:

Step 1 **Advance** u^* using the conservative form:
$$\frac{\partial u^*}{\partial t} + \frac{\partial}{\partial x} f(u^*) = 0 \Rightarrow u^{*n+1} = u^{*n} - \Delta t \cdot \frac{\partial}{\partial x} f(u^{*n})$$

Step 2 **Filter** the result to obtain the updated $\overline{u}$:
$$\overline{u}^{n+1} = G * u^{*n+1}$$

- For initialization at $n = 0$:
  - Reconstruct:
  $$u^{*0} = Q * \overline{u}^0$$
  - Filter:
  $$\overline{u}^1 = G * u^{*1}$$

- At any stage, the updated u^* field satisfies:
$$u^{*n+1} = Q * G * u^{*n+1}$$
since $u^* \approx Q * \overline{u}$, and $\overline{u}^{n+1} = G * u^{*n+1}$, so applying Q again should ideally return u^*, implying:
$$u^* \approx QGu^*$$

- **Subgrid-Scale (SGS) Modeling Viewpoint:**
  - The SGS effect is captured not by explicitly adding an extra term (as in eddy-viscosity models), but by the two-step process of reconstruction (via Q) and filtering (via G).
  - Hence, the SGS modeling here is embedded within the numerical methodology rather than being added explicitly to the equations.

- **Composite Filter Interpretation:**
$$\text{Combined filter:} \quad QG * u^*$$
This composite operator appears repeatedly across time steps:
  - Final operation in one time step is application of G.
  - First step of the next time step begins with Q.
  - Thus, the evolution is mediated by the combined operation QG.

(20250322#217)

Explain why lower order implicit methods are more stable relative to explicit methods:

- A large class of time integration schemes used in numerical analysis are known as **implicit methods**. These methods are defined by updating the solution using future-time information, typically requiring the solution of an algebraic system at each time step.
- For example, a general backward Euler (implicit) method for solving $\frac{du}{dt} = F(u)$ is given by:

  $$u^{n+1} = u^n + \Delta t \cdot F(u^{n+1})$$

  where the function $F(u^{n+1})$ must be evaluated at the future time step. This equation is usually solved iteratively or via matrix inversion.
- **Why don't implicit schemes blow up?**
  - Implicit methods are **unconditionally stable** for many classes of problems, especially stiff equations.
  - Their numerical formulation naturally damps out high-frequency modes, even without an explicit stabilization term.
  - This is because the implicit step inherently incorporates damping through the algebraic solution process.
- **Numerical stabilization effect:**
  - Although numerical schemes (implicit or explicit) introduce discretization errors, implicit schemes tend to introduce **dissipative errors**.
  - These dissipative errors often suppress unstable growth (i.e., instabilities due to discretization or high-frequency modes), which can otherwise lead to blow-up in explicit schemes.
  - Hence, the scheme **stabilizes the computation** by introducing a kind of artificial viscosity or smoothing.
- **Dissipative errors and numerical stability:**
  - Dissipative error refers to artificial decay of amplitude in the numerical solution due to discretization.
  - While this error reduces accuracy, it can enhance numerical robustness by damping unstable components.
  - Therefore, the **numerical error** itself—specifically the dissipative component—acts as a **regularizing mechanism**.
- **Conclusion:** Implicit methods are widely used for their stability benefits. Though they require solving nonlinear systems, their resistance to numerical blow-up and their damping characteristics make them well-suited for stiff or long-time simulations.

(20250322#218)

How is explicit filtering time integration different from that of implicit filtering?

- In numerical methods, it is important to distinguish between:
  - **Implicit methods in time**: where the solution at the next time level involves solving equations that depend on future-time variables (e.g., backward Euler).
  - **Explicit subgrid modeling**: where certain model components, such as filtering or subgrid-scale (SGS) effects, are computed directly from known quantities at the current time.
- In the context of Large Eddy Simulation (LES), we deal with:

  $$\text{Filtered equations:} \quad \overline{u}_t + \overline{u \cdot \nabla u} = -\nabla \overline{p} + \nu \Delta \overline{u} + \tau^{SGS}$$

  where $\tau^{SGS} = \overline{u_i u_j} - \overline{u}_i \overline{u}_j$ is the subgrid-scale stress tensor.
- In models such as eddy viscosity:

  $$\tau^{SGS} \approx -2\nu_t \overline{S}_{ij}$$

  where ν_t is the eddy viscosity (often modeled explicitly), and $\overline{S}_{ij}$ is the resolved strain rate tensor.
- **Key distinction:**
  - These models are not ”implicit” in the numerical time-stepping sense.
  - Instead, they rely on an **explicit modeling step** applied directly to the resolved (filtered) velocity field $\overline{u}$.
  - The subgrid contribution is computed using algebraic formulas (e.g., Smagorinsky model), rather than requiring implicit solution of future-state variables.
- **Explicit Filtering:**
  - Many SGS models involve applying an explicit spatial filter (e.g., convolution with a Gaussian kernel) to the velocity field.
  - This filtering is done outside the time-integration scheme and does not involve solving equations implicitly.
- **Conclusion:**
  - Although the term ”explicit” may refer to both time integration and subgrid modeling, their meanings differ.
  - Here, we are referring to SGS models that are **explicitly defined in terms of known quantities** at the current time step—unlike implicit time-stepping schemes, which require solving coupled equations.

(20250322#219)

How are shocks in compressible flows numerically treated?

- In compressible flow problems, especially those involving shocks, the **shock thickness** is typically of the order of the **mean free path** of the fluid molecules.

- In numerical simulations, it is generally not feasible to use grid spacing as small as the molecular mean free path. As a result, **shock waves appear as jump discontinuities** in computed quantities like velocity, pressure, and density.

- To enable meaningful computation in the presence of discontinuities, it is useful to **recast the governing equations** such that **derivatives act on continuous quantities**.

- Examples of such continuous (or weakly discontinuous) quantities across a shock include:

  - Mass flux: ρu
  - Total enthalpy or stagnation quantities
  - Pressure + momentum flux: $P + \rho u^2$

- In **finite volume methods (FVM)**, the core idea is to compute the evolution of the system by balancing **fluxes across control volumes**:

$$\frac{d}{dt}\int_{\Omega} U \, dx + \int_{\partial\Omega} F(U) \cdot \hat{n} \, dS = 0$$

- The numerical fluxes at cell interfaces need to be treated carefully, especially when discontinuities are present.

- Since **fluxes remain continuous** across shocks, the flux-based discretization helps mitigate non-physical oscillations—but the presence of large gradients near shocks can still lead to numerical instabilities.

- Historically, this was dealt with by introducing **artificial viscosity** into the governing equations to prevent solution blow-up and smear the discontinuity over a few grid points.

- For instance, in first-order upwind schemes, a **numerical diffusion term** is added that mimics a second-order derivative. This stabilizes the solution but introduces diffusion and can reduce accuracy.

- **Inviscid simulations:**
  - In the absence of physical viscosity, the correct method is to use **upwinding** to ensure proper numerical dissipation.
  - For shocks, upwinding ensures the correct entropy condition is satisfied and prevents non-physical oscillations.

- **In viscous flows:**
  - Viscous effects contribute from everywhere in the domain, not just near the shock.
  - Nevertheless, even in viscous computations, the **shock structure** is often under-resolved due to insufficient grid resolution.
  - In practical computations, it is often acceptable to ignore the internal structure of the shock, as long as the numerical method captures the macroscopic effects accurately.

- **Key Goals of Numerical Shock Capturing:**
  - **Correct jump strength:** The method should yield an accurate approximation of the magnitude of the discontinuity.
  - **Correct shock speed:** The shock should propagate at the correct physical speed as dictated by the Rankine-Hugoniot conditions.
- These two aspects form the **functional correctness** of shock-capturing methods, even if the internal shock profile is not resolved.

(20250322#220)

Explain the wave structure and discontinuities in shock tube problem:

- The **shock tube problem** is a classical test case in compressible flow, commonly used to study the propagation of discontinuities such as shock waves, expansion fans, and contact discontinuities.
- Initial configuration consists of:
  - High pressure, high density gas on the left ($x < 0$)
  - Low pressure, low density gas on the right ($x > 0$)
  - A diaphragm at $x = 0$ that separates the two regions initially
- At $t = 0$, the diaphragm is removed, and the system evolves under the Euler equations. The result is a self-similar solution composed of:
  - A **shock wave** propagating to the right into the low pressure region.
  - An **expansion fan (rarefaction wave)** propagating to the left into the high pressure region.
  - A **contact discontinuity** between the two regions of flow that have passed through the shock and the expansion wave, respectively.
- The contact discontinuity separates:
  - Fluid that has passed through the shock (on the right side of the contact line)
  - Fluid that has passed through the expansion wave (on the left side of the contact line)
- Across the contact discontinuity:
  - **Pressure and velocity remain continuous**
  - **Density exhibits a jump**
- The final wave structure (from left to right) is as follows:
  1. Undisturbed high-pressure gas
  2. Expansion fan (smoothly varying region)
  3. Constant state (post-expansion)

4. Contact discontinuity
5. Constant state (post-shock)
6. Shock wave
7. Undisturbed low-pressure gas

- This structure is key for testing numerical solvers for compressible flow, especially for verifying correct shock capturing, entropy consistency, and wave speed resolution.

(20250322#221)

Why is it better to use spectral methods to calculate derivatives as compared to, let's say, finite difference method?

- To understand the behavior of numerical schemes, especially their accuracy and efficiency, it is useful to analyze them in the context of **periodic functions**.
- Let $f(x)$ be a periodic function with period L. It can be expanded in terms of its Fourier series:

$$f(x) = \sum_{k=-N/2}^{N/2} \hat{f}(k) \exp\left(\frac{2\pi i k x}{L}\right)$$

- The derivative of $f(x)$ can be expressed as:

$$\frac{df}{dx} = \sum_{k=-N/2}^{N/2} \left(\frac{2\pi i k}{L}\right) \hat{f}(k) \exp\left(\frac{2\pi i k x}{L}\right)$$

- If we compute the derivative using this spectral representation **explicitly**, the operation scales as $O(N^2)$, since each Fourier mode requires a sum over all points.
- However, using the **Fast Fourier Transform (FFT)** algorithm, we can compute the Fourier coefficients and their derivatives in $O(N \log N)$ operations.
- This provides a **significant gain in both accuracy and speed**, particularly for large N, motivating the use of Fourier methods for periodic domains.
- **Reason for better accuracy:**
  - In traditional finite difference methods (FDM), the derivative at a point depends only on a few nearby points (the stencil).
  - Increasing the number of neighboring points improves accuracy, but still uses local information.
  - In spectral methods, the derivative at a point depends on all Fourier coefficients, which in turn are determined by values at *all* points.

  - Thus, spectral methods incorporate information from the **entire domain**, allowing significantly better accuracy, particularly for smooth functions.

- Hence, spectral methods achieve **global accuracy**, as opposed to local approximations used in finite difference schemes.

(20250322#222)

Explain the idea behind modified wavenumbers and resolution error:

- The accuracy of numerical differentiation schemes can be analyzed using the concept of **modified wavenumbers**.

- This idea is discussed extensively in the work of Lele (1992), *"Compact finite difference schemes with spectral-like resolution"*, Journal of Computational Physics.

- Consider a function $f(x)$ that is periodic and represented as a sum of Fourier modes:

$$f(x) = \sum_k \hat{f}(k) e^{ikx}$$

- The true spatial derivative in the Fourier domain is:

$$\frac{df}{dx} \longleftrightarrow ik\hat{f}(k)$$

- When a numerical differentiation scheme (such as finite difference or compact schemes) is applied to $f(x)$, the Fourier modes are modified. That is, the numerical derivative corresponds to:

$$\frac{df}{dx} \approx ik_{\text{mod}}(k)\hat{f}(k)$$

  where $k_{\text{mod}}(k)$ is called the **modified wavenumber**.

- The function $k_{\text{mod}}(k)$ deviates from the true wavenumber k, especially for high wavenumbers. This leads to two types of errors:
  - **Dispersion error:** Phase speed of waves is incorrectly predicted due to deviation in real part of k_{mod}.
  - **Dissipation error:** Amplitude of waves is damped (artificially), reflected by a nonzero imaginary part of k_{mod}.

- **Resolution error:** arises when the number of grid points N used to represent a function is not sufficient to capture the higher wavenumber components (shorter wavelengths).
  - In particular, spectral methods assume that the function is sufficiently smooth and the grid sufficiently fine to resolve all wavenumbers present in the signal.

  - If this condition is not met, higher frequency components are either aliased or completely missed.

- **Representation error:** refers to the inability of the chosen numerical scheme (including spectral) to faithfully represent all modes of the function due to limited resolution or inherent approximation in the scheme.
- Thus, even in spectral methods, which have exponential accuracy for smooth functions, the effective resolution is limited by:
  - The number of grid points N
  - The highest wavenumber $k_{\max} \sim \pi/\Delta x$ that can be resolved without aliasing
- The use of modified wavenumber analysis provides a unified way to quantify how closely a numerical scheme approximates the true derivative operator across different frequencies.

(20250322#223)

Illustrate how modified wavenumber for a second order derivative case results in error if the base function has higher frequency components:

- Consider a second-order centered finite difference approximation for the derivative of a function $f(x)$:
$$D_2 f(x) = \frac{f(x+\Delta x) - f(x-\Delta x)}{2\Delta x}$$
- Let $f(x)$ be represented by its Fourier series:
$$f(x) = \sum_{k=-N/2}^{N/2} \hat{f}(k)\, e^{2\pi i k x/L}$$
- Applying the second-order finite difference operator D_2 to this expansion gives:
$$D_2 f(x) = \sum_{k=-N/2}^{N/2} \hat{f}(k)\, e^{2\pi i k x/L} \cdot \frac{e^{2\pi i k \Delta x/L} - e^{-2\pi i k \Delta x/L}}{2\Delta x}$$
- Using the identity $e^{i\theta} - e^{-i\theta} = 2i\sin(\theta)$, this becomes:
$$D_2 f(x) = \sum_{k=-N/2}^{N/2} \hat{f}(k)\, e^{2\pi i k x/L} \cdot \frac{i\sin(2\pi k \Delta x/L)}{\Delta x}$$
- Therefore, the numerical wavenumber (or **modified wavenumber**) for the finite difference scheme is:
$$\tilde{k}(k) = \frac{\sin(2\pi k \Delta x/L)}{\Delta x}$$

- In contrast, the true derivative corresponds to a wavenumber:

$$k_{\text{true}} = \frac{2\pi k}{L}$$

- The discrepancy between $\tilde{k}(k)$ and k_{true} causes a spectral error. For small values of k, $\tilde{k}(k) \approx k_{\text{true}}$, but as k increases, the sine term diverges from linear behavior.

- This mismatch leads to significant error at higher wavenumbers. Typically:
  - The approximation is good for wavenumbers up to roughly 1/3 of the Nyquist limit.
  - Beyond that, the sine term underestimates the true wavenumber, leading to dispersion and phase errors.

- Therefore, if a function has substantial energy in the high-frequency (high-wavenumber) modes (e.g., at $k \sim 2/3$ of Nyquist), a second-order finite difference method will poorly approximate its derivative.

- **Conclusion:** Finite difference (FD) methods can incur spectral errors, especially at high wavenumbers. These errors can accumulate and manifest as numerical instability or poor accuracy.

- To reduce spectral error, one must:
  - Choose higher-order schemes or compact schemes.
  - Ensure that the numerical wavenumber $\tilde{k}$ closely matches the true wavenumber k.
  - Increase grid resolution (i.e., going from N to $2N$ points reduces the error at each mode).

- For spectral methods:
  - Derivatives are exact in the Fourier sense.
  - As long as the Fourier representation captures the spectral content of the function, the approximation is accurate.

- For DNS (Direct Numerical Simulation) codes, which aim to resolve all relevant scales, the choice of derivative scheme and grid resolution must together ensure minimal spectral error.

(20250322#224)

Explain how finite difference schemes inherently introduce filtering effects:

- Consider the derivative of a periodic function $f(x)$, represented in Fourier space:

$$f(x) = \sum_{k=-N/2}^{N/2} \hat{f}(k) e^{2\pi i k x / L}$$

- The exact spectral derivative is:

$$\frac{df}{dx} = \sum_{k=-N/2}^{N/2} \left(\frac{2\pi ik}{L}\right) \hat{f}(k) e^{2\pi ikx/L} \quad \Rightarrow \quad \hat{f}_x(k) = ik\hat{f}(k)$$

- For spectral methods, this relation holds exactly. Hence, we write:

$$\hat{f}_x = ik\hat{f}(k), \quad \text{with } \tilde{k}(k) = k$$

- In contrast, finite difference schemes yield an approximation:

$$\hat{f}_x = i\tilde{k}(k)\hat{f}(k)$$

  where $\tilde{k}(k)$ is the **modified wavenumber** depending on the scheme used and grid spacing Δx.

- Define the transfer function:

$$H(k) = \frac{\tilde{k}(k)}{k}$$

  This ratio quantifies how the numerical derivative deviates from the exact derivative.

- Observe that applying a finite difference derivative is equivalent to:
  - Taking the spectrally accurate derivative $ik\hat{f}(k)$,
  - Then multiplying by $H(k) = \tilde{k}(k)/k$.

- This means the finite difference derivative acts as a **spectrally accurate derivative** followed by a **filtering operation**, where $H(k)$ behaves like a low-pass filter.

- For low wavenumbers (large scales), $\tilde{k}(k) \approx k \Rightarrow H(k) \approx 1$, so the derivative is accurate.

- For high wavenumbers (small scales), $H(k) < 1$, leading to attenuation — this is equivalent to filtering out high-frequency modes.

- Hence, **finite difference schemes inherently introduce filtering effects**.

- This is particularly relevant for LES (Large Eddy Simulation):
  - In LES, explicit filtering is often used to separate resolved and subgrid scales.
  - But numerical schemes (e.g., finite difference, upwinding, etc.) can act as **implicit filters**.

- **Implication:** Implicit LES (ILES) relies on this filtering effect from the numerical scheme itself to regularize the equations without adding explicit subgrid-scale models.

- In summary, taking a finite difference derivative is equivalent to:

$$\text{Spectral derivative} \times \text{Low-pass filter}$$

  which naturally suppresses small-scale fluctuations, mimicking the behavior of a filtered LES formulation.

(20250322#225)

What are compact difference schemes?

- Compact finite difference schemes (also called **implicit difference schemes** or **Pade schemes**) are finite difference methods designed to achieve higher spectral accuracy.
- In contrast to **explicit difference schemes**, where the derivative at a point is computed directly from function values:

  $$f_x(x_i) = \sum_{k=i_1}^{i_2} \alpha_k f(x_k)$$

  compact schemes relate the derivative at a point to derivatives at neighboring points as well, resulting in an implicit system:

  $$\sum_{j=j_1}^{j_2} \beta_j f_x(x_j) = \sum_{i=i_1}^{i_2} \alpha_i f(x_i)$$

- This leads to a **linear system of equations** for the unknown derivatives $f_x(x_j)$, which must be solved at each time step or spatial sweep.
- These systems are not boundary conditions; rather, they are interior equations requiring specialized treatment (e.g., modified coefficients) at boundaries.
- By solving this system, we obtain approximations of derivatives that have significantly improved **spectral resolution** compared to explicit methods.
- **Why this works:**
  - Compact schemes implicitly incorporate information from more grid points than explicit stencils of the same size.
  - More global information is encoded in the derivative, enhancing resolution of a wider range of wavenumbers.
  - They minimize dispersion and dissipation errors, crucial for wave propagation and turbulence problems.
- These schemes also help reduce **truncation error**, which refers to the difference between the numerical derivative and the true derivative, and typically scales as a power of the grid spacing Δx.
- In Fourier space, if a filtering operation is represented as:

  $$\hat{G}, \quad \text{then a deconvolution operator } \hat{Q} \text{ satisfies } \hat{Q}\hat{G} = \hat{I}$$

  where $\hat{I}$ is the identity operator. Compact schemes help recover such accuracy in physical space.
- **Example: 4th-order compact difference scheme**

  $$\alpha F_{i-1} + F_i + \alpha F_{i+1} = \beta_{-1} f_{i-1} + \beta_0 f_i + \beta_{+1} f_{i+1}$$

  where:

- $F_i \approx f_x(x_i)$ is the numerical derivative.
  - The left-hand side contains implicit dependence on neighboring derivatives.
  - The right-hand side involves the function values.

- In matrix-vector form, for the entire grid:

$$A\vec{F} = B\vec{f}$$

  where A is a tridiagonal (or banded) matrix formed by α coefficients, and B is formed by the β coefficients. This system can be solved efficiently.

- The interior of the domain uses the same stencil (same band of coefficients), while **endpoint formulas** are modified to accommodate boundary treatment.

- **Conclusion:** Compact schemes deliver higher accuracy and improved spectral properties without requiring very large stencils. They are particularly well-suited for simulations involving smooth wave propagation, turbulence, and acoustic waves.

Example case:

- Consider the compact finite difference formula for the first derivative at grid point i:

$$\alpha F_{i-1} + F_i + \alpha F_{i+1} = \beta_{-1} f_{i-1} + \beta_0 f_i + \beta_{+1} f_{i+1}$$

- Typical 4th-order accurate coefficients are:

$$\alpha = \frac{1}{4}, \quad \beta_{-1} = \frac{3}{4\Delta x}, \quad \beta_0 = 0, \quad \beta_{+1} = -\frac{3}{4\Delta x}$$

- Let $\vec{F} = [F_1, F_2, \ldots, F_N]^T$, and similarly $\vec{f} = [f_1, f_2, \ldots, f_N]^T$. Then the scheme becomes:

$$A\vec{F} = B\vec{f}$$

- Matrix $A \in \mathbb{R}^{N\times N}$ is tridiagonal with:

$$A = \begin{bmatrix} 1 & \alpha & 0 & \ldots & 0 \\ \alpha & 1 & \alpha & \ddots & \vdots \\ 0 & \alpha & 1 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \alpha \\ 0 & \ldots & 0 & \alpha & 1 \end{bmatrix}$$

- Matrix $B \in \mathbb{R}^{N\times N}$ is also tridiagonal:

$$B = \frac{3}{4\Delta x} \begin{bmatrix} 0 & -1 & 0 & \ldots & 0 \\ 1 & 0 & -1 & \ddots & \vdots \\ 0 & 1 & 0 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & -1 \\ 0 & \ldots & 0 & 1 & 0 \end{bmatrix}$$

- The boundary rows of A and B must be modified using lower-order one-sided formulas or ghost points, depending on the application.
- Solve for $\vec{F}$ using:
  $$\vec{F} = A^{-1}B\vec{f}$$
  This gives a vector of approximate derivatives at each grid point with fourth-order spectral-like accuracy.

(20250324#226)

Why isn't increasing grid spacing not always a feasible thing to do?

Let's say we have N points in one direction. Doubling the number of points $\implies 2N$ points in that direction. Total number of points in the domain would be $(2N)^3 = 8N^3$. Although it increases accuracy, it comes at the cost of huge increase in the number of additional floating point operations performed as compared to the N^3 grid (8 fold).

(20250324#227)

How does numerical scheme's order of accuracy affect resolution characteristics?

In LES, the Navier-Stokes equations are spatially filtered to resolve large eddies, and the accuracy of this resolution depends on the numerical scheme's order of accuracy (e.g., finite difference or finite volume methods).

Order of accuracy refers to the truncation error in the Taylor series expansion of the discretization scheme.

Resolution Characteristics: Higher-order schemes reduce numerical diffusion and dispersion, better capturing the steep gradients and small-scale structures of turbulent eddies. For example, using first order forward difference, we'll have high numerical dissipation, and will smear out eddies. This effect would be lessened if we were to use higher order schemes like, say central difference, where is the error is second order $O(\Delta x^2)$.

(20250324#228)

Why is there a need to handle block boundaries in LES?

LES often uses domain decomposition, splitting the computational domain into blocks (subdomains) for parallel processing or complex geometries.

Block boundaries correspond to the interfaces between subdomains where data (i.e. velocity, pressure) must be exchanged or interpolated. Numerical schemes must maintain continuity and accuracy across these boundaries. Low-order schemes may introduce discontinuities or errors (e.g., mismatched fluxes), while higher-order schemes require sophisticated boundary treatments (e.g., ghost cells, interpolation).

LES of turbulent flows are sensitive to boundary effects (e.g., shear layer development). Poor handling can introduce artificial instabilities or dampen resolved eddies near block interfaces.

To take care of these issues, special algorithms (e.g., overlapping grids, conservative interpolation) are needed, increasing complexity but ensuring physical consistency in shear flow simulations.

(20250324#229)

Why does compact differences offer better resolution and lower truncation errors as compared to explicit schemes?

Unlike standard finite differences (e.g., explicit 2nd-order central difference using $u_{i+1} - u_{i-1}$), compact system solves a system (e.g., tridiagonal) to compute the derivatives implicitly, achieving higher accuracy with fewer grid points.

For example, standard 4th order scheme requires a wider stencil (e.g., 5 point) as opposed to compact 4th order scheme (e.g., 3 points) with implicit correction.

Lower truncation errors are due to compact schemes achieving higher-order accuracy (e.g., 4th or 6th) with $O(\Delta x^4)$ or better, reducing the leading order term compared to explicit methods.

Resolution errors are minimized as a result of better spectral resolution - they accurately resolve wider range of wavenumbers per grid point. This is quantified by the modified wavenumber in Fourier analysis: Standard schemes underestimate high-wavenumber components (small eddies). Compact schemes preserves these, crucial for LES where resolved scales must extend close to the SGS cutoff.

(20250324#230)

Why is parallel programming inevitable for LES?

For large and complex geometries, one often requires domain decomposition to effectively compute the flow solution in LES. Each block can be conveniently processed by a set of processes. This can be done in parallel acrosos multuiple processors (e.g., via MPI or OpenMP), block boundaries are dealt with via interprocess communication. Simluating flow in a large domain (e.g., 10^6-10^8 points) and across thousands of timesteps require parallelization to finish in reasonable time (e.g. days vs years).

(20250324#231)

Plot the variation of k^2 for different values of coefficient ν_{SGS} with k. Show SGS effects as well.

(20250324#232)

Why does simple eddy viscosity suppress some of the large scale dynamics?

In LES, the Navier-Stokes equations are filtered to resolve large eddies, while SGS models approximate the effects of smaller, unresolved scales. A simple eddy viscosity model assumes the SGS stress tensor behaves like a viscous term, $\tau_{ij}^{SGS} = -2\nu^{SGS}\,\overline{S}_{ij}$ where ν^{SGS} is the SGS eddy viscosity and $\overline{S}_{ij}$ is the filtered strain rate. The issue is that applying a uniform ν^{SGS} across all scales introduces artificial dissipation that can dampen large-scale dynamics (e.g., coherent structures like vortices in a shear layer). Over-dissipation occurs because the simple eddy viscosity model assumes a purely dissipative energy transfer from resolved to unresolved scales, neglecting backscatter and local flow variations. This leads to excessive damping of resolved-scale dynamics, altering the energy cascade and distorting flow physics.

(20250324#233)

What are some ways in which Dynamic Smagorinsky is better than simple eddy-viscosity or standard Smagorinsky models?

The dynamic Smagorinsky model improves on the standard version by computing ν^{SGS} locally and dynamically, rather than using a fixed value. It uses a test filter (wider than the LES grid filter) to compare resolved stresses at different scales, adjusting the Smagorinsky coefficient C_s via a least-squares approach (Germano procedure). This dynamic version as opposed to the standard Smagorinsky model adapts better to varying flow conditions (e.g., shear flows vs. isotropic turbulence), reducing excessive damping of large scales.

(20250324#234)

What is the issue with standard Smagorinsky model that calls for the use of dynamic Smagorinsky model, particularly in the context of errors in large scales?

The standard Smagorinsky model sets $\nu^{SGS} = (C_s\Delta)^2\,|\overline{S}|$, where C_s is a constant (e.g., 0.1 or 0.2), Δ is the filter width (grid size), and $|S| = \sqrt{2\overline{S}_{ij}\overline{S}_{ij}}$ is the strain rate magnitude. The term $\nu^{SGS}\partial^2\overline{u}/\partial x^2$ in Fourier space becomes $-k^2\nu^{SGS}\hat{\overline{u}}(k)$, dissipating energy proportional to k^2.

The issue is that if we were to use a fixed value of C_s (and through it, ν^{SGS}), it is as if we're assuming a universal dissipation rate. It may fit for some flows, like channel flow with strong shear, but it may fail in other scenarios, like homogeneous isotropic turbulence (HIT), where turbulence lacks directional preference.

(20250324#235)

What modification can be applied to eddy viscosity model so that it doesn't suppress large scale dynamics?

Instead of applying eddy viscosity uniformly, restrict it to small scales, leaving large-scale dynamics (resolved eddies) unaltered. This avoids the suppression of large scales. It preserves the physical behavior of large eddies (e.g., entrainment in shear flows) while modeling only the SGS effects, aligning with LES's goal of resolving significant turbulent structures.

(20250324#236)

How does high pass filtering in eddy viscosity model leads to large scale dynamics being not suppressed by the model?

We use high pass filtering isolates small-scale components of the filtered velocity $\overline{u}$, then we apply eddy viscosity only to those scales. The high-pass filter removes low-wavenumber (large-scale) components, and retains high-wavenumber (small-scale) fluctuations. In practice, this could be a spectral filter or a difference between the full field and a low-pass filtered field.

Eddy viscosity term of viscous dissipation is modeled as $\nu^{SGS}\partial^2\overline{u}/\partial x^2$. In Fourier space, the second derivative becomes $-k^2\hat{\overline{u}}(k)$, where $\hat{\overline{u}}$ is the Fourier transform of $\overline{u}$, and k is the wavenumber.

The result is that the $\nu^{SGS}\left(-k^2\right)\hat{\overline{u}}(k)$ dissipates energy proportional to k^2, but only in the small scales.

The advantage in following this method is that the selective dissipation mimics the physical SGS energy transfer in turbulent shear flows (e.g., jets), where small scales dissipate energy while large scales drive the flow.

(20250324#237)
What happens for $E_1(k)$, $E_2(k)$ and $E_3(k)$ as compared against SGS effects?

(20250324#238)
Plot the comparison of $\tilde{k} = k$ against 2nd order explicit, 6th order explicit, 6th order compact, and 8th order compact difference schemes.

(20250324#239)
Plot the variation of increase in the number of grid points for LES against the result from DNS as a function of wavenumber k.

(20250324#240)
Plot and explain the variation of skewness with wavenumber k for LES with different number

of grid points and the DNS case.

(20250324#241)
For cannonical, self-preserving turbulent flow, state the relation between U_0/U_c and x/D. What is the expected value for B_u?

(20250324#242)
Explain the varying energy spectrum behavior with x/D for a cannonical turbulent round jet case. When does the spectrum broaden and equilibrate?

(20250324#243)
Plot the variation of Evs fD/U_0 for the same turbulent round jet.

(20250324#244)
Plot the centerline development of U_0/U_c vs x/D. Explain why the curving away of one of the lines.

(20250324#245)
Plot the self-similarity profile for the turbulent round jet case. Describe the matching observed.

(20250324#246)

What is done at the outlet boundary of a flow domain in an incompressible flow around a bluff body??

For steady laminar incompressible flow over an airfoil:

- Inflow velocity: U_∞
- Grid requirements:
  - Clustering of grid points near bluff body surface based on Re
  - Resolve boundary layer gradients

Numerical Treatment

- Steady assumption: $\partial^2(\cdot)/\partial x^2 = 0$
- Unsteady artifacts:
  - Naturally arising numerical unsteadiness (unless it is forcibly suppressed)
  - Vortex shedding at trailing edge
- Wake convection:

  $$\frac{\partial}{\partial t}\left(4U_c\frac{\partial(\cdot)}{\partial x}\right) = 0 \tag{54}$$

  where U_c is artificial convection velocity to remove turbulent structures at the outlet.

Key Implementation Aspects

- Frozen vortex method for downstream transport
- Selective damping of numerical oscillations
- Balance between physical accuracy and stability

(20250324#247)

Give a brief overview as to what is done to handle inlet and outlet boundaries in LES:

- **Incompressible flow**:
  - No wave equation admitted

  - Pressure-velocity coupling through Poisson equation

- **Compressible flow**:
  - Always contains pressure waves (acoustic)
  - Information propagates at characteristic speeds $U \pm c$
  - have waves carrying vorticity, energy, and species concentrations

Boundary Condition Formulation

For compressible flow around bluff bodies:

$$\text{Wave speeds} = \begin{cases} c + U_c & \text{(downstream)} \\ U_c - c & \text{(upstream)} \end{cases} \tag{55}$$

Implementation requirements:

- Non-reflecting boundaries:
$$\frac{\partial}{\partial t} + (U_c + c)\frac{\partial}{\partial x} = 0 \quad \text{(outflow)} \tag{56}$$
- Inlet specification for RANS (k-ϵ model):
$$\overline{u} = U_\infty \tag{57}$$
$$k = \frac{3}{2}(U_\infty I)^2 \tag{58}$$
$$\epsilon = C_\mu^{3/4}\frac{k^{3/2}}{L} \tag{59}$$

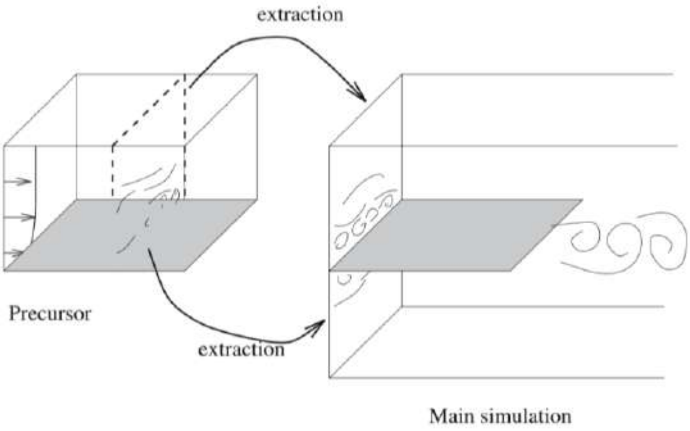
Computational Domain Considerations

- Artificial boundaries must not introduce new physics
- Wave propagation must be unimpeded exiting domain
- Characteristic-based boundary treatments preferred

(20250324#248)

Why use precursor simulations used in simulations of turbulent flows?

To generate realistic correlated fluctuations in a controlled manner before applying them to a larger domain. The goal is to create physically consistent inflow conditions for a main simulation, especially in cases where natural turbulence development would be computationally expensive.



Note that there is no feedback of information from the second simulation to the precursor simulation if both are done separately. This can be an issue when the second simulation issues a signal (e.g., acoustic wave).

(20250324#249)

Why is a precursor simulation needed instead of simply applying synthetic turbulence at the inflow?

Synthetic turbulence methods (e.g., vortex methods, synthetic eddies) may not perfectly replicate large-scale turbulent structures and spatial correlations observed in real turbulence.

A precursor simulation provides physically consistent turbulence, including:

- Proper energy spectra and turbulent kinetic energy (TKE) distribution
- Realistic correlations between velocity components
- A more natural transition to the main computational domain

(20250324#250)

How does domain size affect correlated fluctuations and how to use precursor simulations for large domains?

If the domain contains a large number of grid points, say a billion points or so, it represents a vast computational region.

Instead of running a full-domain simulation, one can simulate turbulence in a smaller precursor domain and copy it over to populate the larger domain.

However, simply tiling (offset copying) the precursor domain will not work, because it introduces artificial periodicity that is unphysical and the large-scale correlated structures must be continuous and randomized, not abruptly repeated.

(20250324#251)

What are some challenges with regards to precursor simulations that one must be aware of?

Homogeneous, isotropic turbulence (HIT) is commonly used as a precursor because it lacks a mean flow gradient. However, in LES/DNS, where the mean flow doesn't have a gradient, the correlation of turbulent structures can decay over time if not handled properly, because of a lack of production term (or synthetic forcing) that sustains or increases turbulence. This means that simply using HIT without careful adaptation may not sustain long-term turbulence in the main simulation.

(20250324#252)

Why can recycling simulation in precursor give rise to locking with time?

Recycling methods feed turbulence data back into the system over time.

However, if not done carefully, artificial patterns can form, leading to time-periodic locking in the turbulence field.

To prevent this, introduce small random perturbations to prevent a strict repetition of structures or use spatial shifting techniques to avoid exact periodicity in turbulent eddies.

(20250324#253)

What are the minimum flow quantities that we should try to match when using a synthetic inflow method?

- Spatial correlation
- Temporal correlation
- Coherent across a range of scales.

(20250324#254)

What is the point of synthetic inflow turbulence generation methods?

Synthetic inflow turbulence generation methods are used to introduce realistic turbulence at the inlet of the computational domain. The goal is to generate a realistic fluctuating velocity field that mimics the statistical properties of turbulence, ensuring the correct Reynolds stress distribution and coherent structures.

(20250324#255)

How does the vortex-based synthetic inflow turbulence generation method work?

We place point vortices in the inflow plane. These vortices evolve based on Biot-Savart law, which describes the induced velocity at a given point due to a vortex:

$$V = \sum_i \frac{\Gamma_i}{2\pi} \frac{(r - r_i)^\perp}{|r - r_i|^2}$$

where Γ_i is the circulation strength of the i-th vortex. The velocity at any given point is a sum of the contributions from all point vortices. This method is able to produce unsteady, correlated velocity fluctuations that evolve realistically over time.

(20250324#256)

Explain the Reynolds stress based synthetic inflow turbulence generation method:

Directly impose a turbulent velocity field that matches the expected Reynolds stress tensor.

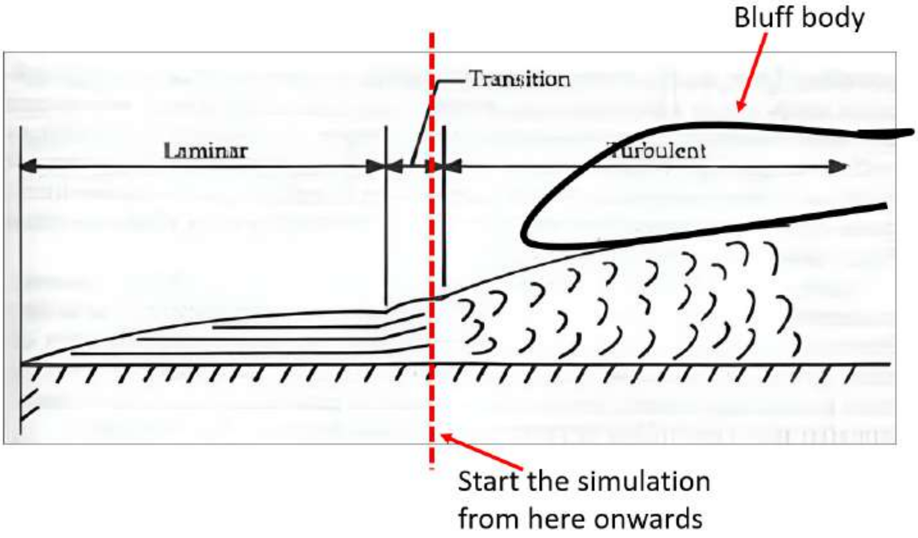
$$R_{ij} = \langle u_i' u_j' \rangle$$

To impose realistic turbulence, we use:

- Random Fourier modes: Generate velocity fields using a random superposition of modes with the correct spectral properties.
- Digital filtering methods: Generate synthetic turbulence by filtering white noise to match a target Reynolds stress distribution.
- Synthetic Eddy Method (SEM): Introduce artificial eddies at the inflow plane to satisfy prescribed turbulence statistics.

(20250324#257)

Let's say we're simulating the interaction between turbulent boundary layer and a bluff body downstream. What are some things that one can do to set up this simulation, without having to compute for the laminar developing region of the boundary layer flow?



The primary interest is in understanding how a turbulent boundary layer (TBL) over a flat plate interacts with a solid bluff body downstream.

The study excludes the initial laminar development region of the boundary layer to save computational resources and focus directly on the turbulent interaction.

To introduce fully developed turbulent boundary layer at the inlet of the computational domain, one can resort to

- precomputed turbulent boundary layer profiles experimental data or DNS/LES.

- synthetic turbulence generation
- precursor simulations, etc.

The velocity profile close to the wall must follow the law of the wall, given by

$$U^+ = \frac{1}{k} \ln y^+ + C$$

where U^+ is the non-dimensional velocity, y^+ is the non-dimensional wall-normal coordinate, κ is the von-Karman constant (~ 0.41), C is an empirical constant (~ 5.0 for smooth walls).

The Reynolds stress components scale as

$$\langle u'v' \rangle \approx -\nu \frac{dU}{dy}$$

The turbulent kinetic energy (TKE) is maximum near the buffer layer and decreases towards the outer layer.

(20250324#258)

What are some of the flow phenomena that can occur as a result of intearction of the turbulent boundary layer with the bluff body?

- Flow separation: Depending on the Re and the bluff body shape, the flow may separate at some location away from the leading edge of the bluff body. The separation point may depend on the incoming turbulence intensity and the adverse pressure gradient introduced by the bluff body.
- Shear layer instabilities: The separated shear layer rolls up into vortices due to Kelvin-Helmholtz instability. These vortices are convected downstream and influence wake turbulence.
- Reattachment and recirculation: If the bluff body is streamlined, the flow may reattach downstream, forming a recirculation bubble. The length of the recirculation region depends on the Reynolds number and the turbulent boundary layer thickness.
- Wake formation and vortex shedding: In the wake region behind the bluff body, vortex shedding occurs, characterized by the Strouhal number:

  $$St = \frac{fL}{U_\infty}$$

  where f is the vortex shedding frequency, L is the characteristic length of the bluff body and U_∞ is the freestream velocity. This vortex shedding leads to pressure fluctuations and aerodynamic forces such as drag and lift oscillations.

- Turbulent Energy Transfer: The interaction of the turbulent boundary layer with the bluff body generates additional turbulent kinetic energy (TKE), modifying the energy cascade in the flow.

(20250328#259)

What is the issue with deciding the amount of clustering near the walls while setting up a computation?

The friction velocity u_τ isn't known a priori, but grid clustering depends on it:

$$u^+ = \begin{cases} y^+ & \text{for } y^+ < 5 \quad \text{(Viscous sublayer)} \\ \frac{1}{\kappa}\ln(y^+) + B & \text{for } y^+ > 30 \quad \text{(Log layer)} \end{cases}$$

where $y^+ = y/\delta_\nu = yu_\tau/\nu$ and $u^+ = U/u_\tau$, $\kappa \approx 0.41$ and $B = 5.0$.

(20250328#260)

What are some effects as a result of the presence of a wall in a turbulent flow?

Kolmogorov scale doesn't dominate near walls (wall effects modify turbulence):

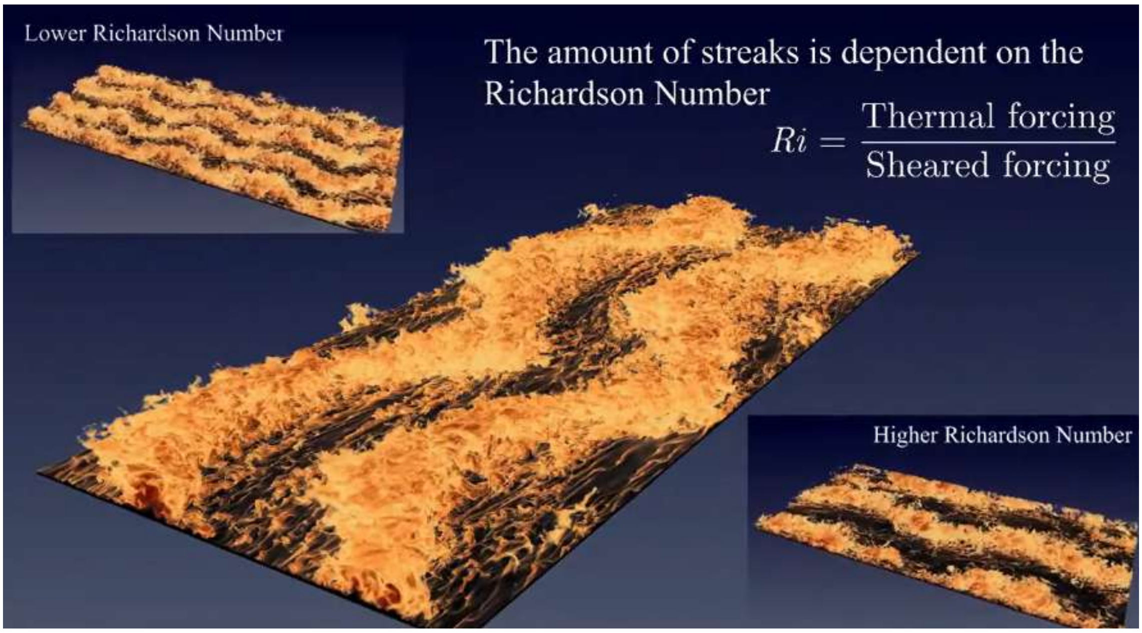
$$\eta_{Kol} = \left(\frac{\nu^3}{\epsilon}\right)^{1/4} \quad \text{but} \quad \epsilon \sim \frac{u_\tau^3}{\kappa y} \quad \text{near walls}$$

(20250328#261)

Briefly describe how one can decide the grid spacing near wall?

Decide grid spacing in wall units based on being able to resolve flow structures like the long-meandering structures near wall. Characteristic streaks have $\lambda_x^+ \approx 1000$, $\lambda_z^+ \approx 100$ (in wall units). Required grid spacing to resolve:

$$\Delta x^+ \approx 50, \quad \Delta z^+ \approx 15, \quad \Delta y^+ \approx 1 \text{ at wall}$$


Lower Richardson Number
The amount of streaks is dependent on the Richardson Number
$Ri = \frac{\text{Thermal forcing}}{\text{Sheared forcing}}$
Higher Richardson Number

(20250328#262)

In short, what is the notion of an accurate solution in a turbulent flow simulation?

To be able to obtain statistically correct flow quantities.

(20250328#263)

How does variation in mean scale have an effect on fluctuating quantities?

The mean velocity profile sets the dominant scales for turbulent fluctuations through mean shear production term:

$$P_{ij} = -\underbrace{\langle u_i' u_k' \rangle}_{\text{Reynolds stress}} \frac{\partial U_j}{\partial x_k} - \langle u_j' u_k' \rangle \frac{\partial U_i}{\partial x_k}$$

Mean shear dominates in viscous sub-layer $y^+ < 5$:

$$\frac{\partial U}{\partial y} \approx \frac{u_\tau^2}{\nu} \implies u_{rms}^+ \sim y^+$$

While in log-layer we have shear decreasing with height

$$\frac{\partial U}{\partial y} = \frac{u_\tau}{\kappa y} \implies u_{rms}^+ \approx 2.5$$

Mean shear can also modify the energy spectrum:

$$E(k) \sim \begin{cases} \epsilon^{2/3} k^{-5/3} & \text{(Isotropic)} \\ (\frac{\partial U}{\partial y})^2 k^{-3} & \text{(Shear)} \end{cases}$$

Region	**Mean Shear Effect**	**Fluctuation Scaling**
Viscous sublayer	$\frac{\partial U}{\partial y} \approx$ const	$u_{rms}^{+} \sim y^{+}$
Buffer layer	Transitional shear	Peak $u_{rms}^{+} \approx 2.8$
Log layer	$\frac{\partial U}{\partial y} \sim 1/y$	$u_{rms}^{+} \approx$ constant

(20250328#264)

How can fluctuating quantities in a turbulent flow influence the mean flow quantities?

If we are able to get the flow structures resolved, then we can get the correlation of fluctuations correctly and that in turn will help in getting the mean fields right as well. The fluctuating quantities influence mean flow quantities in several ways. One such influence is evident through the Reynold's stress term in the RANS equations. Physically the fluctuations help turbulent momentum transport

$$\langle u'v' \rangle \frac{\partial U}{\partial y} < 0$$

typical in shear flows.

(20250328#265)

Why do we vary grid size in wall-normal direction instead of keeping it uniform?

Means vary, so flucuations also vary. There is also change of scales in wall normal direction, as one moves from viscous sub-layer to log-law region and outwards. So the grid size must be appropriately chosen such that it resolves turbulent flow structures pertaining to each such region.

(20250328#266)

How does the grid spacing in DNS and LES vary, lets say for a wall-bounded flow?

Roughly the same wall normal spacing as DNS. Spanwise also they can be roughly kept the same. But in streamwise direction, LES grid spacing can be more relaxed compared to DNS. In DNS, we want the full spectra, so we resolve sufficiently well to capture this spectra for the entire streamwise direction, but in LES, we are interested in large scale structures, and the long structures in streamwise direction and their dynamics can be sufficiently well captured with fewer grid points along that direction.

(20250328#267)

How does the skin friction in LES compare to DNS?

For typical simulations, compared to DNS, skin friction error is roughly around 5%, which can be brought down to 1% using finer grid resolution. But the solution obtained in LES can be taken to be reliable.

(20250328#268)

What are some casees where RANS excels and fails?

RANS is found to perform well for attached flows, but fails when flow separation over a smooth surface is involved. In attached flows having well-developed turbulent boundary layers,

$$\frac{\partial U}{\partial y} \text{ follows log-law}, \quad \nu_t = \kappa u_\tau y$$

Reason: Equilibrium turbulence with established scaling.

It also performs well for fully developed pipe/channel flows (reason: homogeneous turbulence with 1D shear), and in high Reynold's number external aerodynamics (reason: dominant turbulent regions are thin and attached).

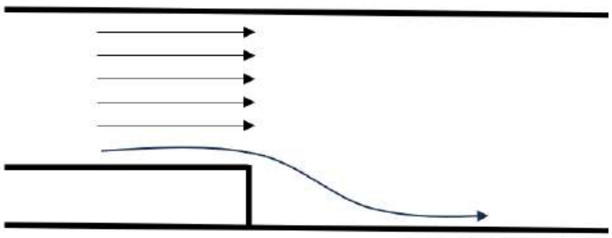
It fails when we have a massive flow separation:

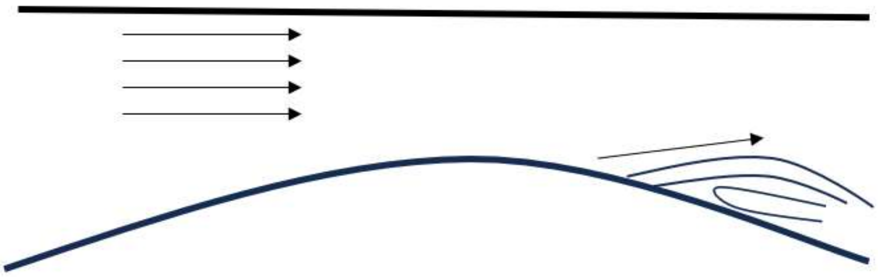
$$\frac{\partial U}{\partial x} \sim \frac{U_\infty}{L} \text{ vs } \nu_t \text{ models assume } \frac{\partial U}{\partial x} \ll \frac{\partial U}{\partial y}$$

Failure Mode: Anisotropic stresses not captured.

In this case RANS shows flow separation like DNS or LES.

But in this case, (like a venturi tube for example), even though in experimental, LES and DNS cases, we observe flow separation, the flow may or may not separate for RANS. Also the separation point need not match with the results from experimental, DNS or LES cases (Note that the separation point need not be a fixed spatial point in time, the separation can happen over a spatial band on the surface bounding the turbulent flow region).





Transient vortex shedding is also not captured well.

$$St = \frac{fD}{U} \text{ (Strouhal number)}$$

Failure Mode: Unsteady phase averaging required.

It is also not good at shock boundary layer interactions

$$\frac{\partial P}{\partial x} \sim \rho u_\tau^2/\delta$$

Failure Mode: Compressibility effects on ν_t unmodeled.

(20250328#269)

Why is LES able to get the separation bubble dynamics correct unlike RANS?

In LES, Reynolds stresses are calculated correctly. We have hence better expectation of getting the flow and hence the bubble dynamics to be correctly captured.

RANS fundamentally misrepresents separation dynamics on smooth surfaces because:

- The modeled Reynolds stress τ^R cannot capture non-equilibrium turbulence
- Eddy viscosity assumptions break down when U/ y→0 U/ y→0

- History effects from upstream development are ignored

For laminar flows,

$$\frac{d\delta}{dx} \sim \frac{1}{Re_x^{1/2}}, \quad \left.\frac{\partial U}{\partial y}\right|_{wall} \sim \frac{U_\infty}{\delta}$$

with separation criterion (Goldstein's singularity),

$$\left.\frac{\partial U}{\partial y}\right|_{y=0} = 0 \text{ when } \frac{dP}{dx} > \frac{\rho\nu^2}{U_\infty^3}\frac{d^3U_e}{dx^3}$$

Based on eddy viscosity assumption, we have

$$\tau_{ij}^R = -\rho\langle u_i' u_j'\rangle \approx \rho\nu_t\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) - \frac{2}{3}\rho k\delta_{ij}$$

Some of the reasons why failure can happen includes $dU/dy \to 0 \implies \tau_{xy}^R \to 0$ being not applied artificially, or that curvature effects $((U_\theta/r)\partial/\partial r(rU_\theta))$ are not taken into account in ν_t models.

(20250328#270)

Give a birds eye view exmaple of applying hybrid RANS-LES:

Take the example of a turbulent flow separating over a smooth surface such as in a venturi tube. We can have hybrid RANS-LES, where we do RANS for incoming flow, and where we expect to see separation and in the separated region, we do LES. Over the flow separation region, we can apply RANS as well. Opting for hybrid means rather than pure LES can cut down on computation cost enormously, but at the same time, its application faces several challenges, such as taking care of boundary conditions. We require turbulent boundary conditions for LES region at the interface in both upstream and downstream locations where we transfer over to RANS. THere should be some way to synthesize usnteady solution from RANS at these boundary locations.

(20250328#271)

Explain briefly about wall modeling for LES:

Without wall modeling, we require to clustering of grid points near the solid surfaces of the flow domain. With wall modeling however, we can use the same grid spacing away from the wall at the near wall locations as well. This can enormously cut down computational costs.

For DNS/ resolved LES without wall modeling,

$$\Delta y^+_{min} \approx 1$$

But with wall modeling,

$$\Delta y^+_{WM} \approx 50 - 100$$

Wall modeling was applied to RANS long before its viability in application to LES was explored. First grid point away from wall is taken to be in log-law region. Since there is no wall tehre, we require to model the shear stress at the first grid point location. We make use of log law to come up with an estimate for shear stress. Here we compare the log law mean statistics to the viscous sub layer region. In LES we know instantaneous quantities acts in log-law layer. So we have $\mu(t)$, $u(t)$ and $\tau_w = \tau_w(t)$, based on $\partial u/\partial y,\ \partial w/\partial y$.

(20250328#272)

Give an example for a small scale phenomena which can happen in a flow which has its effect felt extensively across the flow domain. Explain one method to mitigate the problem of small scales being smaller than smallest flow length scales.

One such example would be that of combustion. Reaction zone thickness in typical flames is usually smaller or around the same order of magnitude as that of Kolmogorov scales. In addition, the flames could be highly wrinkled as well.

If the time scale of the simulation based on the grid scale is larger than that of the flame's associated time scale, it can miss important combustion dynamics such as the dynamical nature of how reactants come into contact with the flame. For example, based on the Damkohler number,

$$\mathrm{Da} = \frac{\tau_{\text{flow}}}{\tau_{\text{chem}}} \begin{cases} \ll 1 & \text{(Chemistry unresolved)} \\ \gg 1 & \text{(Mixing unresolved)} \end{cases}$$

where the typical flow, chemical and mixing time scales are given as

$$\tau_{\text{flow}} = \frac{\Delta}{u'} \quad \tau_{\text{chem}} = \frac{\delta_L}{s_L} \quad \tau_{\text{mix}} = \frac{\lambda^2}{D}$$

So we require these conditions for DNS:

$$\Delta \leq \min(\delta_L, \eta_K), \quad \delta_t \leq \min(\tau_{\text{chem}}, \tau_{\text{Kolmogorov}})$$

For LES,

$$\Delta \approx 5\delta_L, \quad \delta_t \sim \tau_{\text{flow}} \gg \tau_{\text{chem}}$$

One of the adopted solution method is the thickened flame model. Here in the simulation, we thicken the flame while making sure that the flame speed doesn't vary.

$$F = \frac{\delta_L^{LES}}{\delta_L}, \quad D_t = FD, \quad \dot{\omega}_t = \frac{\dot{\omega}}{F}$$

Large reaction rate is typically what causes the reaction zone thickness to become small. So we modify the reaction rate such that the resulting flame thickness is now representable in the LES grid. We then modify the combustion related coefficients to make the now slowed down flame to propagate with the same velocity that it had before. This is analogous to what we do to represent shocks in LES, where we make sure that the shock spans over few cells without changing the shock speed much, when in reality this doesn't happen.

AE226 Assignment-2

Vivek T., 25657, PhD/AE

Questions:

Suppose you wished to analyse turbulent, incompressible flow of air through a duct with a mean velocity $\overline{U} = 75$ m/s and uniform cross-section 0.1 m^2.

(a) Calculate Reynolds number based on mean velocity and diameter D of a circle of the same area.

(b) Using these parameters, as if for flow through a circular pipe, estimate the distance from the wall y mm which corresponds to one wall unit ($y^+ = 1$).

(c) Find the location where the velocity from the viscous sublayer relation matches with that of the log-layer.

(d) Plot on the same graph for $0 < y < D/2$:

(1) Velocity profile using viscous sublayer and log law,

(2) velocity profile using the 1/7-th power law,

$$\frac{U}{U_c} = \left(\frac{y}{D/2}\right)^{1/7}$$

where U_c is the centerline velocity. You may assume $\overline{U}/U_c = 0.85$.

Solutions:

(a) Reynolds number is given by

$$Re = \frac{\rho U D}{\mu}$$

where ρ is the density of the fluid flowing through the domain, U is the characteristic velocity, D is the characteristic length and μ is the dynamic viscosity. Here the fluid of our interest is air, having density $\rho = 1.23$ kg/m^3 under standard conditions for dry air at 25 °C. Dynamic viscosity for air at the same conditions would be 1.849×10^{-5} Pa · s. Characteristic velocity can be taken to be the same as the mean velocity $\overline{U} = 75$ m/s. Characteristic length D can be obtained by equating the uniform cross-section area of the duct with the circular pipe as

$$\text{Circular cross-section area} = \text{Duct cross-section area}$$

$$\implies \frac{\pi D^2}{4} = 0.1 \text{ m}^2$$

or $D = \sqrt{0.4/\pi} = 0.3568$ m.

Hence,

$$\begin{aligned} Re &= \frac{\rho \overline{U} D}{\mu} \\ &= \frac{\left(1.23 \text{ kg/m}^3\right) \cdot (75 \text{ m/s}) \cdot (0.3568 \text{ m})}{(1.849 \times 10^{-5} \text{ Pa} \cdot \text{s})} \\ &= 1.78 \times 10^6 \end{aligned}$$

(b) We use the relation

$$\begin{aligned} y^+ &= \frac{y u_\tau}{\nu} \\ &= \frac{y}{\nu}\sqrt{\frac{\tau_w}{\rho}} \end{aligned}$$

But τ_w is unknown here. To estimate τ_w for the turbulent flow through the circular pipe, we use the skin-friction coefficient relation

$$C_f = \frac{\tau_w}{\frac{1}{2}\rho U^2}$$

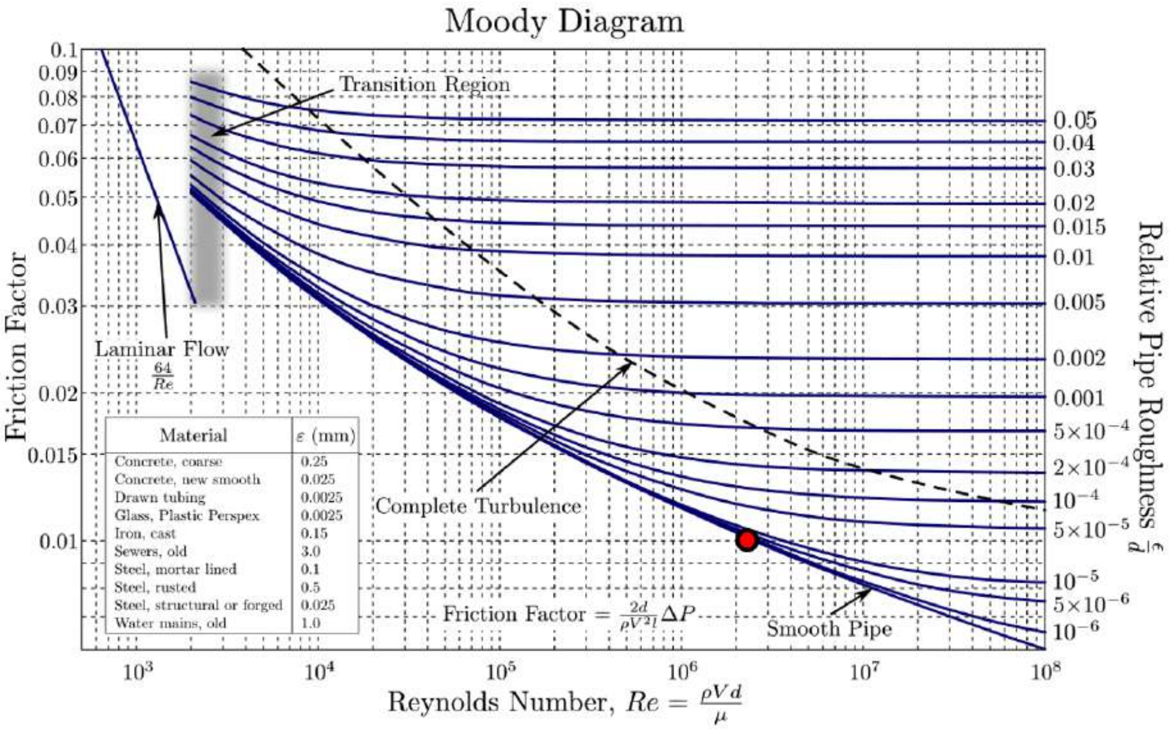
But Darcy friction factor $f = 4C_f$. We can estimate C_f directly using Schlichting's formula assuming smooth pipes, where the relation would be applicable as the obtained Reynolds number Re lies in the range, $10^4 \leq Re \leq 10^9$.

$$C_f = [2\log_{10}(Re) - 0.8]^{-2.3}$$

This gives $C_f \approx 0.0031$. We can also obtain f directly from Moody chart (again assuming smooth pipe) and use it to find C_f. Following the moody chart, we obtain $f \approx 0.01057$ for the obtained $Re = 1.78 \times 10^6$ and $C_f = f/4 \approx 0.0026$ which is roughly 16% different as compared to the result obtained from Schlichting's formula. Since the experimentally obtained C_f and the result obtained from Schlichting's formula matches reasonably well, we go ahead with the Schlichting's formula result for the rest of the analysis.

Using the skin-friction coefficient relation, we obtain τ_w estimate for the assumed smooth circular pipe with turbulent flow

$$\begin{aligned} \tau_w &= \frac{1}{2}\rho \overline{U}^2 C_f \\ &= 0.5 \times 1.23 \times \left(75^2\right) \times 0.0031 \\ &= 10.724 \text{ Pa} \end{aligned}$$



Material	ε (mm)
Concrete, coarse	0.25
Concrete, new smooth	0.025
Drawn tubing	0.0025
Glass, Plastic Perspex	0.0025
Iron, cast	0.15
Sewers, old	3.0
Steel, mortar lined	0.1
Steel, rusted	0.5
Steel, structural or forged	0.025
Water mains, old	1.0

Figure: Moody chart with the point of interest marked with a red circle

Thus $y^+ = 1$ will imply

$$\begin{aligned} y &= \frac{\mu}{\rho}\sqrt{\frac{\rho}{\tau_w}} \\ &= \mu\sqrt{\frac{1}{\rho\tau_w}} \\ &= \left(1.849\times10^{-5}\ \text{Pa}\cdot\text{s}\right)\sqrt{\frac{1}{\left(1.23\ \text{kg/m}^3\right)\cdot(10.724\ \text{Pa})}} \\ &= 5.091033476488723\times10^{-}6\ \text{m} \\ &\quad\text{or}\ \sim 0.0051\ \text{mm}. \end{aligned}$$

(c) The location where the viscous sublayer relation matches with that of the log-layer will lie in the overlap region between $y^+ = 5$ and $y^+ = 30$. The matching location can be obtained by equating the u^+ obtained for the viscous sublayer with the u^+ obtained from the log-layer.

In the viscous sublayer,

$$u^+ = y^+$$

In the log layer,

$$u^+ = \frac{1}{\kappa}\ln\left(y^+\right) + B$$

Equating the two for the same u^+,

$$u^+ = \frac{1}{\kappa}\ln\left(y^+\right) + B$$

This non-linear relation can be solved iteratively using Newton iteration method. Since we have a guess that $y^+ \in [5, 30]$, I chose the midpoint of this range, $y^+ = 17.5$, as the initial guess.

The Newton iteration function used to solve for y^+ in Python:

```python
def find_y_plus_match(y_guess=17.5, k=0.41, B=5.0, tol=1e-6, max_iter=100):
    y_plus = y_guess
    for i in range(max_iter):
        f = y_plus - (1/k)*np.log(y_plus) - B
        df = 1 - (1/k)/y_plus

        y_plus_new = y_plus - f/df

        if abs(y_plus_new - y_plus) < tol:
            return y_plus_new, i+1

        y_plus = y_plus_new
    return y_plus, max_iter
```

The Newton iteration converged to $y^+ = 10.80$ in 4 iterations. Converting from wall units to actual y distance in mm, we have

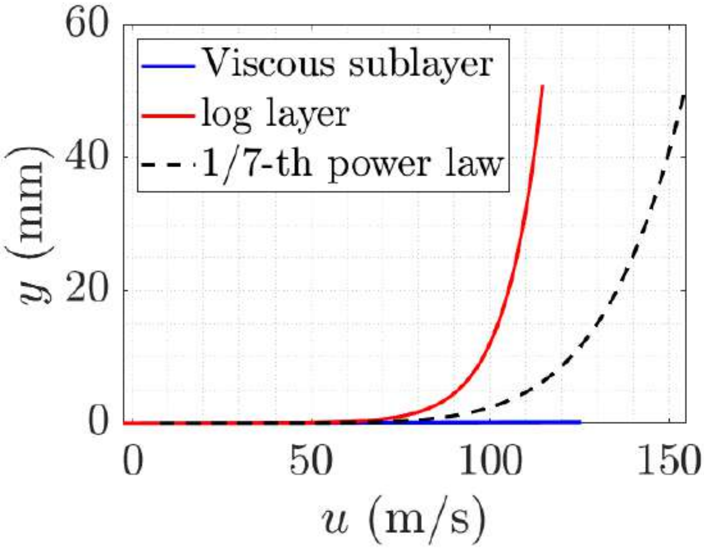
$$y = 10.80 \times 0.0051 = 0.05508 \text{ mm}$$

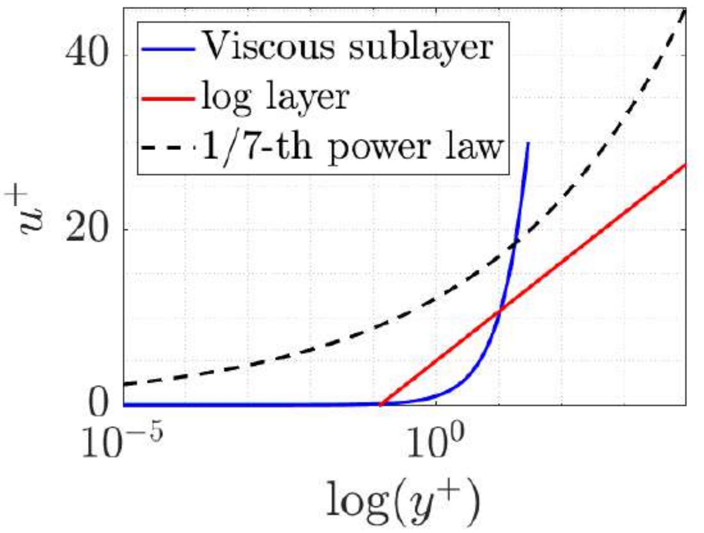
(d) using $\kappa = 0.41$ and $B = 5.0$,

Converting the 1/7-th power law in terms of u^+ and y^+,

$$\begin{aligned}
u^+ u_\tau &= U_c \left(\frac{y^+ \delta_\nu}{D/2} \right)^{1/7} \\
u^+ &= \frac{\overline{U}}{u_\tau} \frac{U_c}{\overline{U}} \left(\frac{y^+ \delta_\nu}{D/2} \right)^{1/7} \\
&= \frac{75}{\sqrt{2\tau_w/\rho}} \frac{1}{0.85} \left(\frac{y^+ \times 0.0051}{0.1784} \right)^{1/7} \\
&= \left(\frac{75}{\sqrt{2 \times 10.724/1.23}} \right) \left(\frac{1}{0.85} \right) \left(\frac{y^+ \times 0.0051}{0.1784} \right)^{1/7} \\
&= 12.716 \times \left(y^+ \right)^{1/7}
\end{aligned}$$

where $u\tau = \sqrt{2\tau_w/\rho} = 4.1758$ m/s.





Question 1(a)

Begin with the expression for production

$$\mathcal{P} = -\langle u_i' u_j' \rangle \frac{\partial U_i}{\partial x_j}$$

of turbulence kinetic energy k in the transport equation for k.

(a) Reduce the expression to that for a turbulent flat plate boundary layer where the mean quantities are independent of the spanwise coordinate and spanwise velocity component also vanishes.

Solution

Given Conditions

For a turbulent flat plate boundary layer:

- Mean flow is statistically 2D (independent of spanwise coordinate z)
- Spanwise mean velocity $W = 0$
- $\frac{\partial}{\partial z} = 0$ for all mean quantities

Step 1: Expand the Production Term

The general production term is:

$$\mathcal{P} = -\langle u_i' u_j' \rangle \frac{\partial U_i}{\partial x_j}$$

Expanding the summation for $i, j = 1, 2, 3$ (where $1 \to x$, $2 \to y$, $3 \to z$):

$$\mathcal{P} = -\left[\langle u'u' \rangle \frac{\partial U}{\partial x} + \langle u'v' \rangle \frac{\partial U}{\partial y} + \langle u'w' \rangle \frac{\partial U}{\partial z} + \langle v'u' \rangle \frac{\partial V}{\partial x} + \langle v'v' \rangle \frac{\partial V}{\partial y} \right] - \left[\langle v'w' \rangle \frac{\partial V}{\partial z} + \langle w'u' \rangle \frac{\partial W}{\partial x} + \langle w'v' \rangle \frac{\partial W}{\partial y} + \langle w'w' \rangle \frac{\partial W}{\partial z} \right]$$

Step 2: Apply Boundary Layer Approximations

- $\frac{\partial}{\partial z} = 0$ for mean quantities
- $W = 0 \Rightarrow$ all derivatives of W vanish
- $\langle u'w' \rangle$ and $\langle v'w' \rangle$ are typically small in 2D boundary layers
- $\frac{\partial U}{\partial x}$ is small compared to $\frac{\partial U}{\partial y}$ in boundary layers

Step 3: Final Reduced Form

The dominant terms remaining are:

$$\mathcal{P} \approx -\langle u'v' \rangle \frac{\partial U}{\partial y}$$

This is the standard form of the production term for a 2D turbulent boundary layer.

Question 1(b)

Using consistent velocity and length scales, show that the velocity gradient tensor $\partial U_i/\partial x_j$ in a thin turbulent boundary layer has only one significant term. Then write the simplified production expression for this flow.

Solution

Scaling Analysis for Boundary Layers

For a turbulent boundary layer with thickness δ developing over a plate of length L, we define:

- Streamwise length scale: $L_x \sim L$
- Wall-normal length scale: $L_y \sim \delta$
- Streamwise velocity scale: U_∞ (freestream velocity)
- Wall-normal velocity scale: V_∞ (from continuity)

Velocity Gradient Tensor Components

The velocity gradient tensor components scale as:

$$\begin{gathered}
\frac{\partial U}{\partial x} \sim \frac{U_\infty}{L} \\
\frac{\partial U}{\partial y} \sim \frac{U_\infty}{\delta} \\
\frac{\partial V}{\partial x} \sim \frac{V_\infty}{L} \\
\frac{\partial V}{\partial y} \sim \frac{V_\infty}{\delta}
\end{gathered}$$

Continuity Requirement

From the continuity equation for incompressible flow:

$$\frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} = 0$$

This implies:

$$\frac{U_\infty}{L} \sim \frac{V_\infty}{\delta} \quad \Rightarrow \quad V_\infty \sim U_\infty \frac{\delta}{L}$$

Order of Magnitude Analysis

Substituting the velocity scales:

$$\begin{aligned}
\frac{\partial U}{\partial x} &\sim \frac{U_\infty}{L} \\
\frac{\partial U}{\partial y} &\sim \frac{U_\infty}{\delta} \\
\frac{\partial V}{\partial x} &\sim \frac{U_\infty \delta / L}{L} = \frac{U_\infty \delta}{L^2} \\
\frac{\partial V}{\partial y} &\sim \frac{U_\infty \delta / L}{\delta} = \frac{U_\infty}{L}
\end{aligned}$$

Comparing terms in the thin boundary layer limit ($\delta/L \ll 1$):

$$\frac{\partial U}{\partial y} \sim \frac{U_\infty}{\delta} \quad \text{(largest term)}$$

$$\frac{\partial U}{\partial x}, \frac{\partial V}{\partial y} \sim \frac{U_\infty}{L} \quad \text{(smaller by factor of } \delta/L\text{)}$$

$$\frac{\partial V}{\partial x} \sim \frac{U_\infty \delta}{L^2} \quad \text{(negligible, order } (\delta/L)^2\text{)}$$

Simplified Production Term

The only significant term in the velocity gradient tensor is $\partial U/\partial y$. Therefore, the production term reduces to:

$$\mathcal{P} = -\langle u'v' \rangle \frac{\partial U}{\partial y}$$

This is the standard form used in boundary layer theory, where:

- $\langle u'v' \rangle$ is the Reynolds shear stress
- $\frac{\partial U}{\partial y}$ is the mean velocity gradient

Question 1(c)

Using Prandtl's mixing length model for Reynolds stress, demonstrate that the production term $\mathcal{P}$ is always positive in turbulent boundary layers.

Solution

Prandtl's Mixing Length Model

The Reynolds shear stress is modeled as:

$$\langle u'v' \rangle = -\nu_t \frac{\partial U}{\partial y}$$

where ν_t is the eddy viscosity. Prandtl's mixing length theory expresses this as:

$$\nu_t = l_m^2 \left| \frac{\partial U}{\partial y} \right|$$

where l_m is the mixing length. Thus:

$$\langle u'v' \rangle = -l_m^2 \left| \frac{\partial U}{\partial y} \right| \frac{\partial U}{\partial y}$$

Production Term Expression

From part (b), the production term is:

$$\mathcal{P} = -\langle u'v' \rangle \frac{\partial U}{\partial y}$$

Substituting the mixing length model:

$$\mathcal{P} = -\left(-l_m^2 \left| \frac{\partial U}{\partial y} \right| \frac{\partial U}{\partial y} \right) \frac{\partial U}{\partial y} = l_m^2 \left| \frac{\partial U}{\partial y} \right| \left(\frac{\partial U}{\partial y} \right)^2$$

Sign Analysis

Notice that:

- $l_m^2 > 0$ (always positive)
- $\left|\frac{\partial U}{\partial y}\right| \geq 0$ (magnitude is non-negative)
- $\left(\frac{\partial U}{\partial y}\right)^2 \geq 0$ (square term is always non-negative)

Final Result

Therefore, the production term simplifies to:

$$\mathcal{P} = l_m^2 \left|\frac{\partial U}{\partial y}\right|^3 \geq 0$$

The production is:

- Strictly positive when $\frac{\partial U}{\partial y} \neq 0$
- Zero only when $\frac{\partial U}{\partial y} = 0$

This confirms that turbulent kinetic energy production is always non-negative under Prandtl's mixing length hypothesis, with the following physical interpretation:

- Energy is always extracted from the mean flow ($\mathcal{P} > 0$)
- No backscatter occurs in this model
- The magnitude depends cubically on the velocity gradient

Question 1(d)

Using the velocity profiles for the viscous sub-layer and logarithmic law region, derive expressions for the production term $\mathcal{P}$ in each layer and demonstrate that production must peak near the wall.

Solution

Wall Coordinates and Velocity Scales

Define wall coordinates:

$$y^+ \equiv \frac{y u_\tau}{\nu}, \quad u_\tau \equiv \sqrt{\frac{\tau_w}{\rho}}$$

where u_τ is the friction velocity and τ_w is the wall shear stress.

1. Viscous Sublayer ($y^+ < 5$)

Velocity Profile

$$U^+ = y^+$$

or in dimensional form:

$$U = \frac{u_\tau^2}{\nu} y$$

Velocity Gradient

$$\frac{\partial U}{\partial y} = \frac{u_\tau^2}{\nu}$$

Reynolds Stress

In viscous sublayer, turbulent fluctuations are suppressed:

$$\langle u'v' \rangle \approx 0$$

Production

$$\mathcal{P} = -\langle u'v' \rangle \frac{\partial U}{\partial y} \approx 0$$

2. Log-Law Region ($30 < y^+ < 0.1\delta^+$)

Velocity Profile

$$U^+ = \frac{1}{\kappa} \ln y^+ + B$$

where $\kappa \approx 0.41$ is von Kármán's constant and $B \approx 5.0$.

Dimensional gradient:

$$\frac{\partial U}{\partial y} = \frac{u_\tau}{\kappa y}$$

Reynolds Stress

In the log layer, Reynolds stress is approximately constant:

$$\langle u'v' \rangle \approx -u_\tau^2$$

Production

$$\mathcal{P} = -\langle u'v' \rangle \frac{\partial U}{\partial y} \approx u_\tau^2 \left(\frac{u_\tau}{\kappa y} \right) = \frac{u_\tau^3}{\kappa y}$$

Production Peak Near the Wall

Comparing both regions:

- **Sublayer**: $\mathcal{P} \approx 0$
- **Log-layer**: $\mathcal{P} \sim 1/y$ (decreases with y)

The transition between these regions ($5 < y^+ < 30$) must contain the peak production because:

- Production increases from zero in the viscous sublayer
- Reaches maximum where turbulent fluctuations become significant
- Then decays as $1/y$ in the log layer

Dimensional Analysis

Expressed in wall units:

$$\mathcal{P}^+ \equiv \frac{\mathcal{P}\nu}{u_\tau^4} = \begin{cases} 0 & \text{in viscous sublayer} \\ \frac{1}{\kappa y^+} & \text{in log layer} \end{cases}$$

The peak production typically occurs around $y^+ \approx 10 - 15$, which is the buffer layer between viscous and log-law regions.

(matthews2006jfm#1)

Why is the propagation and growth of instabilities in the outer fluid merely a kinematic response of the perturbation at the shear layer?

The term kinematic response refers to the way disturbances in the outer fluid behave as a result of perturbations originating at the shear layer, without implying that the outer fluid itself is inherently unstable. It emphasizes that the outer fluid's reaction is a passive consequence of the instability in the shear layer rather than an independent instability mechanism developing within the outer fluid.

When a perturbation grows within the shear layer due to velocity gradients, it does not remain confined to this region. Instead, it extends into the outer fluid, where disturbances appear to propagate and amplify. However, this behavior does not indicate that the outer fluid is unstable on its own; rather, it reflects how disturbances from the shear layer influence the surrounding flow.

This distinction is important in stability analysis because it clarifies that while the outer fluid may show apparent growth of perturbations, it is merely responding to the instability mechanisms in the shear layer. The outer fluid does not generate its own instability but instead passively reacts to the dynamics imposed by the shear layer.

(matthews2006jfm#2)

Why do waves with a real k travel with constant amplitude in the x direction and decay exponentially in the z-direction?

When the streamwise wavenumber k is real, the perturbation represents a traveling wave with no exponential growth or decay in the x-direction. This is because a real k leads to a purely oscillatory solution in x, meaning the disturbance propagates without amplification or attenuation in that direction.

Mathematically, a perturbation in a two-dimensional flow can be expressed as:

$$\psi(x, z, t) = \widehat{\psi}(z)e^{i(kx-\omega t)}$$

where k is the wavenumber in the streamwise direction, ω is the frequency, and $\widehat{\psi}(z)$ represents the shape of the perturbation in the cross-stream direction.

- When k is real, the exponential term eikxeikx represents a traveling wave that oscillates but does not grow or decay in xx, leading to constant amplitude propagation in that direction.
- In the z-direction, however, the wave behavior is determined by the nature of the eigenfunctions. For spatially bounded flows, the solutions must satisfy the boundary

conditions at large z, typically requiring that the disturbances decay as $z \to \infty$. This results in exponentially decaying perturbations in the z-direction.

Physically, this means that the disturbances remain localized around the shear layer and do not radiate energy indefinitely into the outer fluid. Instead, the wave energy remains concentrated near the flow interface, leading to a structure where the disturbance propagates downstream (in x) with a constant amplitude while its influence diminishes in the normal direction (in z).

(matthews2006jfm#3)

Why does the case with complex k have travelling waves with constant amplitude at an angle to the x-axis and exponential decay perpendicular to it?

(matthews2006jfm#4)

For a wake flow, if i were to put a point impulse at a location, won't the waves propagate in all directions? Then why is it that for complex k, the waves travel at an angle to the x axis and have exponential decay normal to this direction?

(matthews2006jfm#5)

How is the impulse response of a shear layer in a wake region equivalent to point source response?

(matthews2006jfm#6)

What is meant by a branch cut?

A **branch cut** is:

- A curve in the complex plane $\mathbb{C}$ across which a multi-valued function is discontinuous
- Introduced to define single-valued branches of functions like:

$$\sqrt{z}, \quad \log z, \quad z^\alpha \ (\alpha \notin \mathbb{Z})$$

- Example: For $\sqrt{z}$, typically placed on $(-\infty, 0]$

Mathematically:

$$f(z) = \sqrt{z} = |z|^{1/2} e^{i\theta/2}, \quad \theta \in (-\pi, \pi]$$

discontinuous across $\theta = \pm\pi$.

(matthews2006jfm#7)

In the complex k_x plane, ξ has branch points at $k_x = ik_y$ and $k_x = \pm\infty$. Why?

A $k_x = ik_y$, $\xi = \sqrt{k_x^2 + k_y^2} = 0$, meaning ∞ wavelength. Mathematically, if we take $\xi = f(z) = \sqrt{z}$, where $z = \sqrt{k_x^2 + k_y^2}$, we already know that $z = 0$ is a branch point, as starting from an arbitrary z, if we trace a closed path which encloses $z = 0$ and comes back to the original position in the complex z plane, $f(z)$ would now have a different value → multi-valued function.

(matthews2006jfm#8)

Why do we have a second set of branch points along the k_i axis?

(matthews2006jfm#9)

Why is that the most unstable perturbation is found when the wavenumber along the direction in which the bulk flow velocity is lower is 0?

(matthews2006jfm#10)

Why does the sign function pin the branch cuts to the k_i axis? Why will retaining ξ make it possible for branch cuts to be shifted?

(matthews2006jfm#11)

Why do we have to shift the integration path such that it doesn't pass through any poles or branch points of $g(k)$? We are integrating

$$\frac{B}{(\partial D/\partial \omega)[k, \omega(k)]} e^{-igt} dk$$

and not $g(k)$ itself. So why care about $g(k)$'s poles and branch points? Does $g(k)$ having poles and branch points at some k mean the term above will also have poles and branch points at that point? Also why not care about the poles and branch points of the term itself, rather than $g(k)$. Does $g(k)$'s poles and branch points match with those of the term above?

(matthews2006jfm#12)

What are eigenfunctions?

(matthews2006jfm#13)

What does eigenfunctions diverging mean?

(matthews2006jfm#14)

How does avoiding poles and branch points of $g(k)$ while integrating from $k = -\infty$ to ∞ result in avoiding plane of diverging eigenfunctions?

(matthews2006jfm#15)

Why does the common approach of considering long time limit, where dominant contribution towards the integration term in perturbation stream function comes from the neighborhood of highest g_i on the integration path work?

(matthews2006jfm#16)

Why is the growth rate of this form?

$$g \equiv [\omega(k) - i\xi z/t - kx/t]$$

(matthews2006jfm#17)

Why is the surface for the grwoth rate $g(k)$ hyperbolic everywhere?

(matthews2006jfm#18)

Why is it that if the surface of $g(k)$ is hyperbolic, then path on k always passes through one ore more saddle points?

(matthews2006jfm#19)

What is a hyperbolic surface?

(matthews2006jfm#20)

When we compute RANS equations, we get the Reynold's stress term. This term arises out of non-linear term in the NS equation. But the terms in Re stress are just covariance terms, so they're only capturing the linear interaction between u_i' and u_j'. How is this possible? Is the RANS equation linear? Does this mean that the mean flow is independent of the non-linear interaction between the velocity fluctuations?

(matthews2006jfm#21)

Instead of applying averaging for each of the terms in RANS equation, can we apply mutual information?

(matthews2006jfm#22)

Is mutual infromation operation linear with the derivative term?

(matthews2006jfm#23)

Are RANS equations applicable only in stationary flows?

(matthews2006jfm#24)

Why does the wavecrests travel in one direction and then grow and decay in perpendicular directions?

For a localized wavepacket with exponential growth/decay:

- Wavecrests propagate along phase velocity direction ($\mathbf{v}_p \parallel \mathbf{k}$)
- Amplitude varies perpendicular to $\mathbf{k}$
- Constant growth rate contours align with wavefronts

Mathematical Justification

Consider a complex wavevector $\mathbf{K} = \mathbf{k} + i\kappa$ with:

$$\begin{aligned} \psi(\mathbf{x}, t) &= A \exp(i\mathbf{K} \cdot \mathbf{x} - i\omega t) && (60) \\ &= \underbrace{A \exp(-\kappa \cdot \mathbf{x})}_{\text{Growth/decay}} \underbrace{\exp(i\mathbf{k} \cdot \mathbf{x} - i\omega t)}_{\text{Wave propagation}} && (61) \end{aligned}$$

Physical Explanation

The alignment occurs because:

1. **Phase propagation**: Requires $\mathbf{v}_p \parallel \mathbf{k}$ since:

$$\phi = \mathbf{k} \cdot \mathbf{x} - \omega t = \text{constant} \implies \mathbf{v}_p = \frac{\omega}{|\mathbf{k}|}\hat{\mathbf{k}} \tag{62}$$

2. **Growth localization**: The imaginary component κ must be perpendicular to $\mathbf{k}$ to maintain:

$$\mathbf{K} \cdot \mathbf{K} = k^2 - \kappa^2 + 2i\mathbf{k} \cdot \kappa = \text{real (for physical solutions)} \tag{63}$$

   which requires $\mathbf{k} \cdot \kappa = 0$.

3. **Eigenmode structure**: The dominant mode selects κ normal to wavefronts for maximal spatial localization.

Consequences

- Wavepacket energy propagates along **k**-direction
- Amplitude modulation occurs perpendicular to propagation
- Group velocity may differ from phase velocity direction