# Information Theory

## Guide to AI and Data Courses · Part V: Mathematical Foundations

Entropy | Compression | Channel capacity | Error-correcting codes | Measures in ML | Neighbouring courses

Sercan KÜLCÜ · Giresun University · Computer Engineering

© Sercan Külcü. All rights reserved. · Interactive HTML version of these slides: https://sercankulcu.github.io/files/ai_courses/slides/Chapter_34_Information_Theory.html

INTRODUCTION

# What Question Does It Ask?

> **How is information measured, and how is it compressed and transmitted with the least loss?**

Information theory measures the information in a message by probability: an unlikely event carries more information, and the average information is called **entropy**. The course derives the limits on how far data can be compressed and how fast it can be sent reliably over a noisy channel.

### At a glance

| | | |
|---|---|---|
| **APPROACH**<br>measuring information with probability | **INPUT**<br>source and channel model | **OUTPUT**<br>bounds, codes, compression rate |
| **SUCCESS MEASURE**<br>bits, channel capacity | **MATH LOAD**<br>high: probability | **DATA NEEDS**<br>none |

INTRODUCTION

# A Short History

**1928**
Hartley proposes measuring information by the logarithm of the number of possible messages.

**1948**
Shannon defines entropy and channel capacity in "A Mathematical Theory of Communication".

**1951**
Kullback and Leibler publish a divergence measure between two probability distributions.

**1952**
Huffman publishes the algorithm that builds the prefix code with minimum average length.

**1977**
Lempel and Ziv publish the dictionary-based LZ77 compressor, ancestor of ZIP and PNG.

**1993**
Turbo codes arrive: a practical code comes within about half a decibel of the Shannon limit.

THE COURSE

# Core Concepts

**Entropy**
Average information per symbol of a source, in bits.

**Conditional entropy**
Uncertainty left in one variable once the other is known.

**Mutual information**
How much two variables tell about each other.

**KL divergence**
Extra cost of coding one distribution with a code built for another.

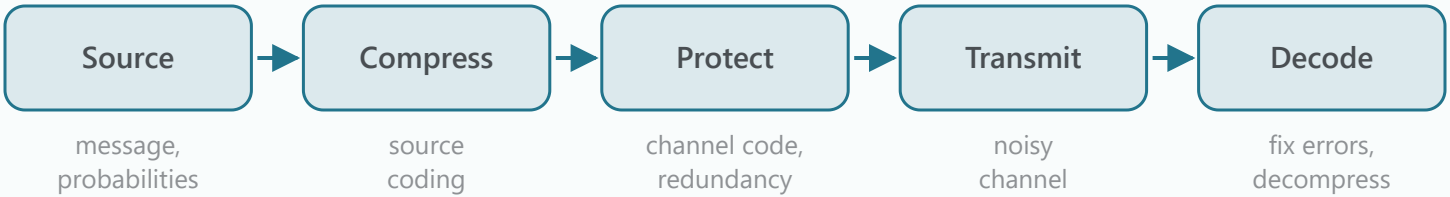
**Cross-entropy**
The most common loss function in classification.

**Source coding**
Lossless compression: Huffman, arithmetic coding, LZ.

**Channel capacity**
Highest rate that can pass reliably through a noisy channel.

**Error-correcting codes**
Adding redundancy to detect and correct errors.

# A Typical Workflow




**Example:** A photo sent from a phone is first compressed with JPEG; the wireless channel code then adds controlled redundancy. Bits corrupted in the air are corrected with that redundancy at the receiver and the photo opens. Compression and protection look like opposites, yet both come from the same theory.

THE COURSE

# What Does the Course Cover?

1. Entropy, joint and conditional entropy
2. Mutual information and KL divergence
3. Jensen's and the data processing inequalities
4. Typical sets and the AEP
5. Source coding theorem and Kraft inequality
6. Huffman codes and arithmetic coding
7. Universal compression: Lempel-Ziv
8. Channel capacity and the channel coding theorem
9. Differential entropy and continuous variables
10. The Gaussian channel and Shannon-Hartley
11. Rate-distortion theory and lossy compression
12. Error-correcting codes: Hamming, LDPC

## Skills you will gain

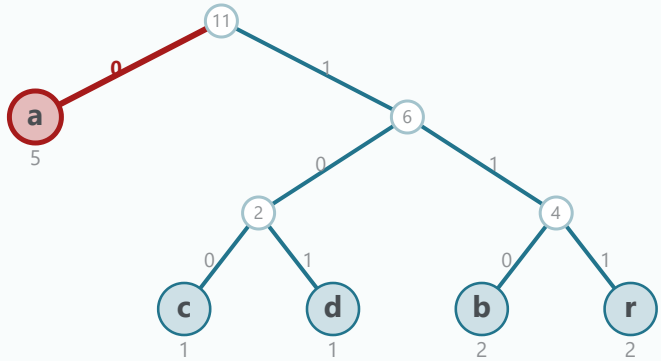
computing entropy and mutual information · writing a Huffman coder · measuring compression with zlib · computing channel capacity · writing proofs with inequalities

A typical semester; the order and coverage vary between universities.

TRY IT

# Send the Message in Fewer Bits

A battery-powered sensor sends its messages by radio, and every bit costs energy. Type a text or pick a preset: the Huffman tree gives frequent letters short codes. Click a letter in the tree to see its code.




Bits sent (letter by letter):

**0**1101110**0**1000**1**0**1**0**1**1011**1**0

Text to send

abracadabra

**abracadabra** | tongue twister | repetitive | all equal

Letter “a” occurs 5 times; its code is **0** (1 bits).

5 distinct letters, 11 letters
Entropy H = **2.04** bits/letter
Huffman average L = **2.09** bits/letter
8-bit code: 88 bits → Huffman **23 bits** (74% saved)

Information theory sets the limit: no lossless code can be shorter than the **entropy** on average. Huffman stays within one bit of it. When all letters are equally frequent, the saving comes only from the small alphabet.

CONTEXT

# Prerequisites

Courses worth knowing before this one, and why they are needed.

| Course | Why it is needed |
| --- | --- |
| Discrete Mathematics | logarithms, sets, counting and binary representation |
| Algorithms | greedy algorithms and priority queues: building a Huffman code |

CONTEXT

# Are You Ready for This Course?

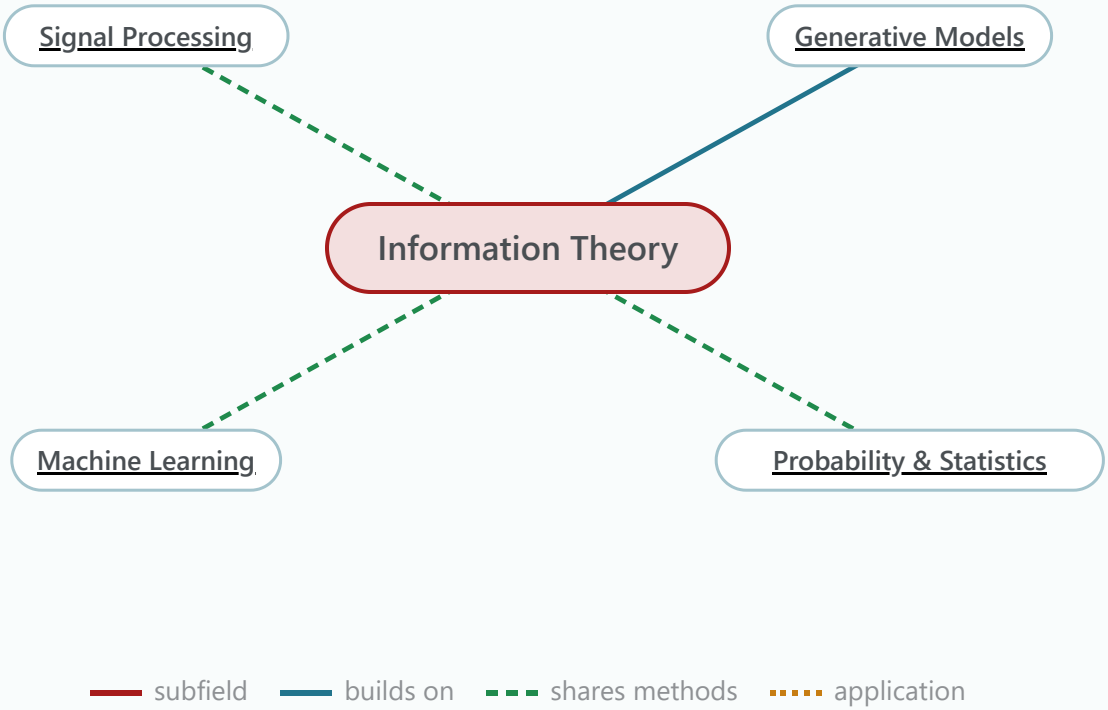
Ask yourself; for every "no", the right-hand column shows where to start.

| Ask yourself | If not, start with |
|---|---|
| Can you use the rules of logarithms and compute in base 2? | Logarithms from high school maths |
| Can you explain conditional probability and expected value? | Probability and Statistics |
| Can you write a number in binary and tell how many bits it needs? | Number bases from discrete mathematics |
| Can you write a greedy algorithm that uses a priority queue? | Greedy methods and heaps from an algorithms course |

If you can say "yes" to most of these, you are ready; one or two gaps can be closed during the semester.

CONTEXT

# Neighbouring Courses



OVERLAP

# Overlapping Courses

| Relation | Course | Shared ground |
|---|---|---|
| builds on | Generative Models builds on this course | KL divergence, likelihood |
| shares methods | Shares methods with Machine Learning | cross-entropy loss, decision tree split criterion |
| shares methods | Shares methods with Digital Signal Processing | compression, sampling, channels |
| shares methods | Shares methods with Probability and Statistics | entropy, mutual information |

# Where It Differs (1/2)

| Criterion | Information Theory | Probability & Statistics | Signal Processing |
|---|---|---|---|
| **Core question** | How is information measured, and how is it compressed and transmitted with the least loss? | How is uncertainty measured, and how do we draw reliable conclusions from limited data? | How are sampled signals analyzed, filtered and transformed? |
| **Approach** | measuring information with probability | probability models and inference | mathematical transforms and filters |
| **Input** | source and channel model | sample data | sampled signal |
| **Output** | bounds, codes, compression rate | estimate, confidence interval, test result | filtered signal, spectrum |
| **Success measure** | bits, channel capacity | unbiasedness, power, significance | signal-to-noise ratio, frequency response |
| **Math load** | high: probability | the theory itself | high: complex numbers, Fourier |
| **Data needs** | none | a sample | no learning; the signal suffices |

- **Probability & Statistics:** Probability and statistics describe distributions and uncertainty; this course derives information content and transmission limits from them.
- **Signal Processing:** Signal processing filters and transforms a signal; this course asks how much information it can carry and how fast.

# Where It Differs (2/2)

| Criterion | Information Theory | Machine Learning | Generative Models |
|---|---|---|---|
| **Core question** | How is information measured, and how is it compressed and transmitted with the least loss? | How does a program generalize from examples? | How does a model generate new samples that look like real data? |
| **Approach** | measuring information with probability | statistical learning | learning the data distribution |
| **Input** | source and channel model | labeled or unlabeled examples | huge, mostly unlabeled data |
| **Output** | bounds, codes, compression rate | a model that makes predictions | new samples: images, text, audio |
| **Success measure** | bits, channel capacity | error on unseen data | realism and diversity (FID etc.) |
| **Math load** | high: probability | high: linear algebra, probability, optimization | high |
| **Data needs** | none | moderate to large | very large |

- **Machine Learning:** ML uses cross-entropy as a loss function; this course explains where that measure comes from and what it means.
- **Generative Models:** Generative models minimize KL divergence in training; this course studies the divergence itself and its link to code length.

PRACTICE

# Applications

**File compression**
ZIP, gzip and PNG: lossless compression.

**Image and audio**
JPEG and MP3: lossy compression.

**Mobile networks**
Turbo codes in 4G, LDPC and polar codes in 5G.

**Data storage**
Error-correcting codes in QR codes, CDs and SSDs.

**Machine learning**
Cross-entropy loss, information gain in decision trees.

**Feature selection**
The most informative features by mutual information.

PRACTICE

# Risks and Ethics

### Mutual information from little data

Histogram estimates of mutual information are biased upward on small samples; an irrelevant feature looks informative.

### The wrong KL direction

KL(p||q) and KL(q||p) give different models: one pushes to cover every mode, the other to lock onto a single mode.

### Secrets leaked by compression

Compressing secret data together with attacker-chosen text lets the size reveal the secret; the CRIME attack on TLS did this.

### Assuming independence

Treating symbols as independent overstates entropy; compression estimates and randomness assessments go wrong.

In depth: Explainable AI and AI Ethics.

PRACTICE

# Tools and Datasets

## Tools and libraries

Python | NumPy | SciPy | zlib | scikit-learn | MATLAB

## Classic example problems

Loaded die and coin toss | Binary symmetric channel | Entropy of English text | Building a Huffman tree | Twenty questions game | Canterbury corpus

Information theory is mostly done with proofs and calculation; code is written to measure entropy and to test compression ratios on real files.

PRACTICE

# Common Misconceptions

| Common belief | In fact |
| --- | --- |
| Information is measured by the meaning of a message. | Shannon information ignores meaning; it measures how unexpected a message is, that is, its probability. |
| A good program can shrink any file. | No lossless method shrinks every input; data cannot be compressed below its entropy. |
| Error-free transmission over a noisy channel is impossible. | Below capacity the error probability can be made as small as desired (Shannon). |
| KL divergence is a distance. | It is not symmetric and violates the triangle inequality: p to q and q to p give different values. |

NEXT STEPS

# After This Course

## Courses that build on or extend this one

Generative Models

## Related careers

Communications engineer | Compression and codec developer | Machine learning researcher | Cryptography specialist | Data storage engineer

Information theory is the mathematical gateway to ML: cross-entropy, KL divergence and mutual information reappear in generative models, decision trees and representation learning. On the communications side, continue with digital signal processing and coding theory.

NEXT STEPS

# Further Reading


- Cover and Thomas, *Elements of Information Theory*, 2nd ed., Wiley, 2006.
- MacKay, *Information Theory, Inference, and Learning Algorithms*, Cambridge University Press, 2003.
- Shannon, "A Mathematical Theory of Communication", *Bell System Technical Journal* 27, 1948.
- Shannon and Weaver, *The Mathematical Theory of Communication*, University of Illinois Press, 1949.

# The End

## Chapter 34: Information Theory

← Previous chapter · Map of the field · Next chapter →