Today

- More on Shannon's theory.
- Contrast with Hamming theory.
- More on Hamming's paper.
- More error-correcting codes.
- Finite fields; Linear codes.
- Codes and their duals.
- Hadamard codes: duals of Hamming codes.

Proof of Converse Coding Theorem

General coding theorem

- General Source: Markovian chain, with each state associated with some symbol.
- Entropy of Markovian distributions; and Rate of Source.
- General Channel: Map from Σ to Γ with probability associated with it.
- Mutual information between distributions and capacity of channel.
- More general channel: Markov chain with edges labelled by pair from $\Sigma \times \Gamma$ and a probability.
- Capacity of such channels.

General Coding Theorem

Mega-Theorem: Every Source has Rate. Every Channel has Capacity. Reliable information transmission (with error going to zero as length of message increases) is possible iff Rate $<$ Capacity.

Some of the main contributions

- Rigorous Definition of elusive concepts: Information, Randomness.
- Mathematical tools: Entropy, Mutual information, Relative entropy.
- Theorems: Coding theorem, converse.
- Emphasis on the "feasible" as opposed to "done".

Contrast with Hamming

- Similar notions of k - message length, and n the block length, and k/n the rate.
- Completely different notation: Shannon focusses the functions E, D, while Hamming doesn't mention either and instead focusses on the set $\{E(x)|x\}$, or the code. Shannon does not mention the code.
- Principal goals different. Hamming seems focussed on adversarial error - making minimum distance the principal objective. Shannon on probabilistic, making Probability of decoding failure the principal objective.

- Shannon non-constructive, while Hamming constructive (reflects maybe on personality, not theory).
- But Hamming theory most critical to Shannon theory as well.
- Prob. decoding failure won't decay exponentially unless min. distance is linear (for avg. codeword).
- Min. distance of codes is easier to reason with, and so codes with large min. distance have been easier to construct.
- Codes with large minimum distance have also (empirically) had low decoding-error probability.

Hamming Goals

- Families of codes (for infinitely many n) with large rate (k/n), large relative distance (d/n), and small alphabet q_n.
- Code is *asymptotically good* if q_n bounded, and $k/n > 0$ and $d/n > 0$. (Take limits over n). First goal is to construct asymptotically good codes. Such codes tolerate $p > 0$ over some q-ary symmetric channel with positive rate.
- Later goal: Construct "optimal" codes (and determine what optimal is!).

Back to Hamming's paper

- Constructed codes with $d = 1, 2, 3, 4$.
- $d = 4$: Add parity check bit to code with odd d and get code with even d.
- Hamming decoding algorithm.
- Suffices to construct a *constant rate* code with polytime encoding + decoding for BSC_p. (Shown by Elias.)
- Hamming lower bound.

Finite fields and linear error-correcting codes

- Field: algebraic structure with addition, multiplication, both commutative and associative with inverses, and multiplication] distributive over addition.
- Finite field: Number of elements finite. Well known fact: field exists iff size is a prime power. See lecture notes on algebra for further details. Denote field of size q by $\mathbb{F}_q$.
- Vector spaces: V defined over a field $\mathbb{F}$. Addition of vectors, multiplication of vector with "scalar" (i.e., field element) is defined,

and finally an inner product (product of two vectors yielding a scalar is defined).

- If alphabet is a field, then ambient space Σ^n becomes a vector space $\mathbb{F}_q^n$.
- If a code forms a vector space within $\mathbb{F}_q^n$ then it is a linear code. Denoted $[n, k, d]_q$ code.

Why study this category?

- Linear codes are the most common.
- Seem to be as strong as general ones.
- Have succinct specification, efficient encoding and efficient error-detecting algorithms. Why? (Generator matrix and Parity check matrix.)
- Linear algebra provides other useful tools: Duals of codes provide interesting constructions.
- Dual of linear code is code generated by transpose of parity check matrix.

Example: Dual of Hamming codes

- Message $\mathbf{m} = \langle m_1, \ldots, m_\ell \rangle$.
- Encoding given by $\langle\langle \mathbf{m}, \mathbf{x} \rangle\rangle_{\mathbf{x} \in \mathbb{F}_2^\ell - \mathbf{0}}$.
- Fact: (will prove later): $\mathbf{m} \neq 0$ implies $\Pr_{\mathbf{x}}[\langle \mathbf{m}, \mathbf{x} \rangle = 0] = \frac{1}{2}$
- Implies dual of $[2^\ell - 1, 2^\ell - \ell - 1, 3]_2$ Hamming code is a $[2^\ell - 1, \ell, 2^{\ell-1}]$ code.
- Often called the simplex code or the Hadamard code. (If we add a coordinate that is zero to all coordinates, and write 0s as -1s, then the matrix whose rows are all the codewords form a $+1/-1$ matrix whose product with its transpose is a multiple of the identity matrix. Such matrices are called Hadamard matrices, and hence the code is called a Hadamard code.)
- Moral of the story: Duals of good codes end up being good. No proven reason.

Next few lectures

- Towards asymptotically good codes:
  - Some good codes that are not asymptotically good.
  - Some compositions that lead to good codes.