Coding Theory

Amit Rajaraman

Summer 2020

Contents

§0. Notation

$\mathbb{N}$ represents the set $\{1, 2, 3, \ldots\}$.
$\mathbb{Z}$ represents the set of integers $\{\ldots, -2, -1, 0, 1, 2, \ldots\}$.
$\mathbb{R}$ represents the set of real numbers.

For $n \in \mathbb{N}$, $[n]$ represents the set $\{1, 2, \ldots, n\}$.

Definition 0.1. An *alphabet* is a finite non-empty set. Elements of an alphabet are typically called *letters* or *symbols*.

An alphabet is usually denoted by Σ. We typically use q to denote $|\Sigma|$.
For $n \in \mathbb{N}$, Σ^n represents the set of length n strings of Σ, that is, the set $\{a_1a_2a_3 \cdots a_n \mid a_i \in \Sigma \text{ for all } i \in [n]\}$.
We also often represent an element of Σ^n as a row vector.
For a set Ω, we denote the power set of Ω by 2^Ω.

Definition 0.2. A *permutation* of a set $S = \{x_1, x_2, \ldots, x_n\}$ is a bijection from S to itself. We denote a permutation f of S by

$$\begin{pmatrix} x_1 & x_2 & \cdots & x_n \\ \downarrow & \downarrow & & \downarrow \\ f(x_1) & f(x_2) & \cdots & f(x_n) \end{pmatrix}$$

Unless mentioned otherwise, assume that $\log = \log_2$.
We assume that the reader is familiar with o, O, ω, and Ω notation used to describe the asymptotic behaviour of functions.

§1. Preliminaries

§§1.1. Metric Spaces

Definition 1.1. A *metric space* is an ordered pair (M, d) where M is a set and $d : M \times M \to \mathbb{R}$ is a *metric* on S, that is, a function such that for all $x, y, z \in M$,

(i) $d(x, y) = 0 \iff x = y$,

(ii) $d(x, y) = d(y, x)$, and

(iii) $d(x, z) \leq d(x, y) + d(y, z)$.

Theorem 1.1. If d is a metric over a set M, then $d(x, y) \geq 0$ for all $x, y \in M$.

Proof. For $x, y \in M$, We have

$$\begin{aligned} d(x, y) + d(y, x) &\geq d(x, x) \\ d(x, y) + d(x, y) &\geq 0 \\ d(x, y) &\geq 0 \end{aligned}$$

Note that equality occurs if and only if $x = y$. ■

§§1.2. Combinatorics

If $n, m \in \mathbb{Z}$ with $0 \leq m \leq n$, the binomial coefficient $\binom{n}{m}$ is defined by

$$\binom{n}{m} = \frac{n!}{m!(n-m)!}$$

where $0! = 1$ and $m! = m(m-1)(m-2)\cdots(2)(1)$ for $m > 0$.

Lemma 1.2. The number of unordered selections of m distinct objects that can be made from a set of n distinct objects is $\binom{n}{m}$.

Theorem 1.3 (Binomial Theorem)**.** Let $x, y \in \mathbb{R}$ and $n \in \mathbb{N}$. Then

$$(x + y)^n = \sum_{i=0}^{n} \binom{n}{i} x^i y^{n-i}.$$

Definition 1.2. A *balanced block design* consists of a set S of v elements, called *points* or *varieties*, and a collection of b subsets of S, called *blocks*, such that for some fixed $k, r, \lambda \in \mathbb{N}$,

(i) each block contains exactly k points,

(ii) each point lies in exactly r blocks, and

(iii) each pair of points occurs together in exactly λ blocks.

Such a design is called a (b, v, r, k, λ)-design.

Example. Take $S = \{1, 2, 3, 4, 5, 6, 7\}$ and the subsets as $\{1, 2, 4\}$, $\{2, 3, 5\}$, $\{3, 4, 6\}$, $\{4, 5, 7\}$, $\{5, 6, 1\}$, $\{6, 7, 2\}$, $\{7, 1, 3\}$. This is a $(7, 7, 3, 3, 1)$-design.

Note that in a balanced block design, $bk = vr$ and $r(k-1) = \lambda(v-1)$.

Definition 1.3. The *incidence matrix* $A = (a_{ij})$ of a (b, v, r, k, λ)-design is a $v \times b$ matrix whose i, jth entry is given by

$$a_{ij} = \begin{cases} 1 & x_i \in B_j \\ 0 & x_i \notin B_j \end{cases}$$

Note that the number of 1s in any column is k and the number of 1s in any row is r.

Example. The incidence matrix corresponding to the example given above is

$$\begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Definition 1.4. A (b, v, r, k, λ)-design is called *symmetric* if $v = b$ and $k = r$. Such a design is referred to as a (v, k, λ)-design.

Definition 1.5. A *Hadamard design* is a $(4t - 1, 2t - 1, t - 1)$-design.

§§1.3. Number Theory

Unless mentioned otherwise, assume that p is a prime.

Theorem 1.4 (Fundamental Theorem of Arithmetic)**.** In $\mathbb{N}$, every number greater than 1 can be represented as a product of prime numbers, and further, this representation is unique up to the order of the factors.

Definition 1.6. The *greatest common divisor* (abbreviated gcd) of two or more numbers not all 0 is defined to be the largest positive integer that divides each of the integers.

The gcd of two integers x, y is denoted (x, y).
If a divides b, we write $a \mid b$.
If p is a prime number and a, r numbers such that $p^r \mid a$.

Lemma 1.5 (Bezout's Lemma)**.** If x and y are nonzero integers and $d = (x, y)$, there exist $\alpha, \beta \in \mathbb{Z}$ such that $\alpha x + \beta y = d$. Furthermore, d is the smallest positive integer that can be represented in the form $\alpha x + \beta y$ where $\alpha, \beta \in \mathbb{Z}$.

If $m \mid (a - b)$ for integers a, b, m, we write $a \equiv b \pmod{m}$.

Definition 1.7. Let a, m be integers. A *modular multiplicative inverse* of a modulo m is an integer x such that $ax \equiv 1 \pmod{m}$.

Theorem 1.6. Let $a, m \in \mathbb{Z}$. The modular multiplicative inverse of a modulo m exists if and only if $(a, m) = 1$.

Proof. We have

$$\begin{aligned} ax \equiv 1 \pmod{m} &\iff ax - 1 = ms \text{ for some } s \in \mathbb{Z} \\ &\iff ax - ms = 1 \text{ for some } s \in \mathbb{Z} \\ &\iff (a, m) \mid 1 \\ &\iff (a, m) = 1 \end{aligned}$$

■

Theorem 1.7 (Stirling's Approximation)**.** For every integer $n \geq 1$, we have

$$\sqrt{2\pi n}\left(\frac{n}{e}\right)^n e^{\lambda_1(n)} < n! < \sqrt{2\pi n}\left(\frac{n}{e}\right)^n e^{\lambda_2(n)}$$

where

$$\lambda_1(n) = \frac{1}{12n+1} \text{ and } \lambda_2(n) = \frac{1}{12n}.$$

§§1.4. Group Theory

Definition 1.8. A group $(G, \cdot)$ is a set G along with a binary operation $\cdot : G \times G \to G$ (We write $\cdot((a, b))$ as $a \cdot b$ for $a, b \in G$) such that

(i) $(a \cdot b) \cdot c = a \cdot (b \cdot c)$ for all $a, b, c \in G$,

(ii) There exists an *identity* element $e \in G$ such that $a \cdot e = e \cdot a = a$ for all $a \in G$ (this identity element is unique, see 1.8 below), and

(iii) For all $a \in G$, there exists an element $b \in G$ (called the *inverse* of a) such that $ab = ba = e$.

A group $(G, \cdot)$ in which $a \cdot b = b \cdot a$ for all $a, b \in G$ is called an *abelian group*.
The identity element of a group written multiplicatively is usually written as 1.

Theorem 1.8. The identity element of a group is unique.

Proof. Let e and e' be identities of a group $(G, \cdot)$. We have $e \cdot e' = e$ as e' is an identity and $e \cdot e' = e'$ as e is an identity. Thus, $e = e'$ and the identity is unique. ■

A common example of a group is $\mathbb{Z}$ under addition.

We define the set $\mathbb{Z}/n\mathbb{Z}$ for some integer n as follows. Let $\sim$ be a relation given by

$$a \sim b \text{ if and only if } n \mid (b - a).$$

It may be shown that $\sim$ is an equivalence relation. Each equivalence class is given by $\overline{a} = \{a + kn \mid k \in \mathbb{Z}\}$. There are precisely n equivalence classes, namely $\overline{0}, \overline{1}, \ldots, \overline{n-1}$. These n equivalence classes are the elements of the set $\mathbb{Z}/n\mathbb{Z}$.
For $\overline{a}, \overline{b} \in \mathbb{Z}/n\mathbb{Z}$, we further define addition and multiplication as

$$\overline{a} + \overline{b} = \overline{a + b} \text{ and } \overline{a} \cdot \overline{b} = \overline{a \cdot b}$$

It may be checked that the above is well-defined.
We see that $\mathbb{Z}/n\mathbb{Z}$ is an abelian group under the addition operation with identity $\overline{0}$ and the inverse of $\overline{a}$ as $\overline{-a}$. We denote this group as $\mathbb{Z}/n\mathbb{Z}$ or $\mathbb{Z}_n$.

We often drop the $\cdot$ and simply write $a \cdot b$ as ab and write the group $(G, \cdot)$ as just G. We also write $aa \cdots a$ (n times) as a^n.

Theorem 1.9. Let G be a group. Then the inverse of any element of the group is unique.

Proof. Let $a \in G$ and b, c be inverses of a. We have $ab = ac = 1$. Premultiplying by b gives $(ba)b = (ba)c$, that is, $b = c$. ■

Definition 1.9. Let G be a group. A subset H of G is a subgroup of G if H is nonempty and it is closed under products and inverses. That is, $a, b \in H$ implies $a^{-1} \in H$ and $ab \in H$. If H is a subgroup of G, we write $H \leq G$.

Note that if $H \leq G$, the identity of G belongs to H as well.

Definition 1.10. If G is a group and $a \in G$, the smallest positive integer n such that $a^n = 1$ is called the *order* of a.

In the above case, the set $\{1, a, a^2, \ldots, a^{n-1}\}$ form a *cyclic* subgroup with a as *generator*. Note that the order of this subgroup is equal to the order of a.

Definition 1.11. Let H be a subgroup of group G. For any $a \in G$, the set $aH = \{ah \mid h \in H\}$ is called a *left coset* or just *coset*. An element of a coset is called a *representative* of the coset.

Theorem 1.10. Let N be a subgroup of a group G. The set of left cosets of H in G partition G. Furthermore, for all $u, v \in G$, $uN = vN$ if and only if $v^{-1}u \in N$.

Proof. First of all, as $N \leq G$, $1 \in N$. Thus $g \in gN$ for all $g \in G$, that is,

$$G = \bigcup_{g \in G} gN$$

To show that distinct left cosets have empty intersection, let $uN \cap vN \neq \varnothing$ for some $u, v \in G$. We must show that $uN = vN$. Let $x \in uN \cap vN$. Then $x = un = vm$ for some $n, m \in N$. This gives $u = v(mn^{-1})$. For any $t \in N$, $ut = v(mn^{-1}t) \in vN$ as $mn^{-1}t \in N$. Thus $uN \subseteq vN$. Similarly, we get $vN \subseteq uN$. Therefore, $uN = vN$ if they have nonempty intersection and we get that the set of left cosets partition G.
By the first part of this theorem, we get $uN = vN$ if and only if $u \in vN$, which is equivalent to $v^{-1}u \in N$. ■

If H is a normal subgroup of group G, the set of cosets of H in G again form a group by defining $(aH)(bH) = (ab)H$. This multiplication makes sense as H is normal. This group is called the *quotient group* and is denoted by G/H.

Theorem 1.11 (Lagrange's Theorem)**.** If H is a subgroup of a finite group G, $|H|$ divides $|G|$ and the number of left cosets of H in G is $\frac{|G|}{|H|}$.

Proof. Let $|H| = n$ and the number of left cosets of H be k. As the set of left cosets partition G, by the map $F : H \to gH$ defined by $h \mapsto gh$ is a surjection from H to the left coset gH. Further, F is injective as $gh_1 = gh_2 \implies h_1 = h_2$. This proves $|gH| = |H| = n$. Since G is partitioned into k subsets each of cardinality n, $|G| = kn$. Thus $k = \frac{|G|}{n} = \frac{|G|}{|H|}$. ■

As a corollary, note that the order of any element of a finite group divides the order of the group.

Theorem 1.12 (Cauchy's Theorem)**.** If G is a finite group and p is a prime dividing the order of G, then G contains an element of order p.

We omit the proof of the above theorem.

§2. Field Theory and Linear Algebra

§§2.1. Introduction to Fields

Definition 2.1. A field $(F, +, \cdot)$ is a set F along with two binary operations $+ : F \times F \to F$ and $\cdot : F \times F \to F$ (We write $+((a, b))$ and $\cdot((a, b))$ as $a + b$ and $a \cdot b$ respectively for $a, b \in F$) such that

(i) $+$ and $\cdot$ are associative. That is, $(a + b) + c = a + (b + c)$ and $a \cdot (b \cdot c) = (a \cdot b) \cdot c$ for all $a, b, c \in F$.

(ii) $+$ and $\cdot$ are commutative. That is, $a + b = b + a$ and $a \cdot b = b \cdot a$ for all $a, b \in F$.

(iii) There exist two distinct elements in F called 0 and 1 such that $a + 0 = a$ and $a \cdot 1 = a$ for all $a \in F$.

(iv) For every $a \in F$, there exists an element in F, denoted $-a$, such that $a + (-a) = 0$.

(v) For every $a \neq 0$ in F, there exists an element in F, denoted a^{-1} or $1/a$, such that $aa^{-1} = 1$.

(vi) Multiplication is distributive over addition, that is, $a \cdot (b + c) = a \cdot b + a \cdot c$ for all $a, b, c \in F$.

The above definition is just equivalent to saying that a field is a set F along with two binary operations $+ : F \times F \to F$ and $\cdot : F \times F \to F$ such that $(F, +)$ is an abelian group with identity 0, $(F \setminus \{0\}, \cdot)$ is an abelian group with identity 1, and multiplication distributes over addition.

We shall often represent a field $(F, +, \cdot)$ as just F and $a \cdot b$ for $a, b \in F$ as just ab.
Common examples of fields are $\mathbb{R}$ and $\mathbb{Q}$.

Theorem 2.1. Let F be a field. For all $a, b \in F$,

(i) $a0 = 0$.

(ii) $ab = 0 \implies a = 0$ or $b = 0$.

Proof.

1. We have $a(0) = a(0 + 0) = a0 + a0$. Adding $-(a0)$ on either side gives the required result.
2. If $a \neq 0$, a has a multiplicative inverse. Then we have $(a^{-1}a)b = a^{-1}0$ which gives $b = 0$. This is the required result.

■

A *finite field* is a field with a finite set of elements. The number of elements in a finite field is called its *order*.

Theorem 2.2. For $n \in \mathbb{N}$, consider the set $\mathbb{Z}_n$ with addition and multiplication defined modulo n, that is, $\overline{a} + \overline{b} = \overline{a + b}$ and $\overline{a} \cdot \overline{b} = \overline{ab}$ for $\overline{a}, \overline{b} \in \mathbb{Z}_n$. $\mathbb{Z}_n$ is a field if and only if n is a prime.

Proof. If n is not a prime, then there exist $a, b \in \mathbb{N}$ both less than n such that $ab = n$, that is, $\overline{a} \cdot \overline{b} = \overline{0}$. As the group $\mathbb{Z}_n$ under addition has identity $\overline{0}$, we see that $\mathbb{Z}_n$ cannot be a field by 2.1.
For prime n, $\mathbb{Z}_n$ is a field as for any $a \notin \overline{0}$, $(a, n) = 1$ and thus a modular multiplicative inverse exists for every element of $\mathbb{Z}_n \setminus \{0\}$ (Recall 1.5). ■

This field, called the *prime field* of order n, is denoted $\mathbb{F}_n$.

Let F be a field. For $a \in F, n \in \mathbb{N}$, we denote $a + a + \cdots + a$ (n times) as na and $aa \cdots a$ (n times) as a^n.

§§2.2. Characteristic of a Field

Definition 2.2. Let F be a field. The smallest positive integer n such that $n1 = 0$ is called the *characteristic* of F and is denoted $\operatorname{char} F$. If no such n exists, we say that F has characteristic 0.

Note that if $\operatorname{char} F = n$, then $na = 0$ for all $a \in F$ ($na = n(1a) = (n1)a = 0$).

Theorem 2.3. Let F be a finite field. Then $\operatorname{char} F$ is prime.

Proof. On the contrary, assume that $n = \operatorname{char} F$ is composite, that is, $n = ab$ for some $a, b \in \mathbb{N}$, $a, b > 1$. We have $n1 = 0$, that is, $(a1)(b1) = 0$. Then 2.1 implies that $a1 = 0$ or $b1 = 0$. As $a, b < n$ and n is the smallest positive integer such that $n1 = 0$, this is a contradiction. Thus, n must be prime. ■

Theorem 2.4. Let F be a finite field. Then the order of F is equal to p^n for some prime p and $n \in \mathbb{N}$.

Proof. Let $\operatorname{char} F = p$. Then since 1 has order p in the group $(F, +)$, p divides the order of F.

Let $q \neq p$ be another prime dividing the order of F. By 1.12, there exists an element of order q in $(F, +)$, that is, there is some non-zero a such that $qa = 0$. We also have $pa = 0$ because $p = \operatorname{char} F$. As p and q are distinct primes, $(p, q) = 1$.

By 1.5, there exist $m, n \in \mathbb{Z}$ such that $mp + nq = 1$. We then have $mp(a) + nq(a) = 1(a)$ which implies $0 = m(pa) + n(qa) = a$. This is a contradiction.

Thus, p is the only prime that divides the order of F. ■

Definition 2.3. Fields F and G are *isomorphic* if there is a bijection $\varphi : F \to G$ such that $\varphi(x + y) = \varphi(x) + \varphi(y)$ and $\varphi(xy) = \varphi(x)\varphi(y)$ for all $x, y \in F$. Such a map is called an *isomorphism*.

Theorem 2.5. Given any prime power q, there exists a unique field of order q (up to isomorphism).

We omit the proof of the above theorem.

Given the above, we unambiguously denote the field of order q as $\mathbb{F}_q$.

§§2.3. Introduction to Linear Algebra

We assume that the reader has an introductory level knowledge of linear algebra and merely state the definitions and theorems without proofs for the sake of completeness.

Definition 2.4. Let $(V, +)$ be an abelian group, $\mathbb{F}$ a field, and let a multiplication $\mathbb{F} \times V \to V$ exist such that

(i) $1\mathbf{a} = \mathbf{a}$ for all $\mathbf{a} \in V$.

(ii) $\alpha(\beta\mathbf{a}) = (\alpha\beta)\mathbf{a}$ for all $\alpha, \beta \in \mathbb{F}$ and $\mathbf{a} \in V$.

(iii) $\alpha(\mathbf{a} + \mathbf{b}) = \alpha\mathbf{a} + \alpha\mathbf{b}$ for all $\alpha \in \mathbb{F}$ and $\mathbf{a}, \mathbf{b} \in V$.

(iv) $(\alpha + \beta)\mathbf{a} = \alpha\mathbf{a} + \beta\mathbf{a}$ for all $\alpha, \beta \in \mathbb{F}$ and $\mathbf{a} \in V$.

Then V is called a *vector space* over $\mathbb{F}$. The identity of $(V, +)$ is denoted by $\mathbf{0}$.

In this case, the elements of V are called *vectors* and the elements of $\mathbb{F}$ are called *scalars*.

Let q be a prime power and $n \in \mathbb{N}$. We denote the vector space $\mathbb{F}_q^n$ over $\mathbb{F}_q$ by $V(n, q)$.

Definition 2.5. Let V be a vector space over $\mathbb{F}$. A non-empty subset W of V is a *subspace* of V if it is a vector space over $\mathbb{F}$ under the same addition and scalar multiplication defined for V.

Theorem 2.6. A non-empty subset W of a vector space V over $\mathbb{F}$ is subspace if and only if $\mathbf{x}, \mathbf{y} \in W \implies \mathbf{x} + \mathbf{y} \in W$ and $\mathbf{x} \in W, \alpha \in \mathbb{F} \implies \alpha\mathbf{x} \in W$.

Definition 2.6. A *linear combination* of r vectors $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_r$ in a vector space V over $\mathbb{F}$ is a vector of the form $a_1\mathbf{v}_1 + a_2\mathbf{v}_2 + \cdots + a_r\mathbf{v}_r$ where $a_i \in \mathbb{F}$ for all valid i.

Definition 2.7. A set of vectors $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_r\}$ is said to be *linearly dependent* if there are scalars a_1, $a_2, \ldots, a_r$ not all 0 such that

$$a_1\mathbf{v}_1 + a_2\mathbf{v}_2 + \cdots + a_r\mathbf{v}_r = \mathbf{0}.$$

Definition 2.8. A set of vectors $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_r\}$ is said to be *linearly independent* if it is not linearly dependent, that is,

$$a_1\mathbf{v}_1 + a_2\mathbf{v}_2 + \cdots + a_r\mathbf{v}_r = \mathbf{0} \implies a_1 = a_2 = \cdots = a_r = 0$$

for scalars $a_1, a_2, \ldots, a_r$.

Definition 2.9. Let V be a vector space and $S = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_r\}$ be a subset of V. S is called a *spanning set* or *generating set* of V if every element of V can be expressed as a linear combination of elements of S.

Definition 2.10. A spanning set of a vector space V which is also linearly independent is called a *basis* of V.

Theorem 2.7. Let V be a vector space. Any spanning set of V contains a basis of V.

Theorem 2.8. Let W be a subspace of vector space $V(n, q)$ and $B = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_k\}$ a basis of W. Then

(i) Every vector in W can be expressed *uniquely* as a linear combination of elements of B.

(ii) W contains exactly q^k vectors.

Definition 2.11. Let V be a vector space and B be a finite basis of V. The number of elements in B is called the *dimension* of V and is denoted $\dim V$. We also then say that V is a *finite-dimensional vector space*. If B is infinite, we say that V is *infinite-dimensional*.

It can be shown that the dimension of a vector space is independent of our choice of basis. Unless mentioned otherwise, assume that any vector space mentioned henceforth is finite-dimensional.

§§2.4. Inner Product Spaces

Definition 2.12. Let V be a vector space over $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$. An *inner product* on V is a function $V \times V \to \mathbb{F}$, given by $(\mathbf{u}, \mathbf{v}) \mapsto \langle \mathbf{u}, \mathbf{v} \rangle$. For all $\mathbf{u}, \mathbf{v}, \mathbf{w} \in V, \alpha \in \mathbb{F}$, it must satisfy the following axioms:

(i) $\langle \mathbf{u}, \mathbf{v} \rangle = \overline{\langle \mathbf{v}, \mathbf{u} \rangle}$ (Hermitian property or conjugate symmetry)

(ii) $\langle \mathbf{u}, \mathbf{v} + \mathbf{w} \rangle = \langle \mathbf{u}, \mathbf{v} \rangle + \langle \mathbf{u}, \mathbf{w} \rangle$ (additivity)

(iii) $\langle \mathbf{u}, \alpha\mathbf{v} \rangle = \alpha \langle \mathbf{u}, \mathbf{v} \rangle$ (homogeneity)

(iv) $\langle \mathbf{v}, \mathbf{v} \rangle \geq 0$ with $\langle \mathbf{v}, \mathbf{v} \rangle = 0 \iff \mathbf{v} = 0$ (positive definite)

An *inner product space* is a vector space with an inner product defined on it.
For example, the dot product defines an inner product on $\mathbb{R}^n$ as a vector space.

Omitting positive definiteness, we extend this idea similarly to vector spaces over finite fields (We take $\overline{x} = x$ for scalar x). Let $\mathbf{u} = u_1u_2\cdots u_n$ and $\mathbf{v} = v_1v_2\cdots v_n$ be elements of $V(n, q)$. The dot product of $\mathbf{u}$ and $\mathbf{v}$ is given by

$$\mathbf{u} \cdot \mathbf{v} = u_1v_1 + u_2v_2 + \cdots + u_nv_n.$$

Definition 2.13. The *norm* of a vector $\mathbf{v}$ in an inner product space V is given by

$$\|\mathbf{v}\| = \sqrt{v \cdot v}.$$

While we state the following definitions and theorems for inner product spaces, they also hold for $V(n, q)$ under the dot product.

Definition 2.14. Let V be an inner product space and $u, v \in V$. If $\mathbf{u} \cdot \mathbf{v} = 0$, we say that $\mathbf{u}$ and $\mathbf{v}$ are *orthogonal* and write $u \perp v$.

Lemma 2.9. For any $\mathbf{u}, \mathbf{v}, \mathbf{w} \in V(n, q)$ and $\alpha, \beta \in \mathbb{F}_q$,

(i) $\mathbf{u} \cdot \mathbf{v} = \mathbf{v} \cdot \mathbf{u}$.

(ii) $(\alpha\mathbf{u} + \beta\mathbf{v}) \cdot \mathbf{w} = \alpha(\mathbf{u} \cdot \mathbf{w}) + \beta(\mathbf{v} \cdot \mathbf{w})$.

Definition 2.15. Let V be an inner product space and W a subspace of V, we define the *orthogonal subspace* of W by

$$W^{\perp} = \{\mathbf{v} \in V \mid \mathbf{v} \perp \mathbf{w} \text{ for all } \mathbf{w} \in W\}.$$

Theorem 2.10. Let V be a finite dimensional inner product space and W be a subspace of V. Then $\dim W + \dim W^{\perp} = \dim V$.

We discuss the proof of a specific form of the above theorem (which is what we require) in 6.10.

Theorem 2.11. Let V be an inner product space and W be a subspace of V. Then $(W^{\perp})^{\perp} = W$.

Proof. We clearly have $W \subseteq (W^{\perp})^{\perp}$. But $\dim(W^{\perp})^{\perp} = n-(n-k) = k = \dim W$ and thus $W = (W^{\perp})^{\perp}$. ■

§§2.5. The Geometric Lemma

Lemma 2.12 (Geometric Lemma)**.** Let $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_m \in \mathbb{R}^n$ be non-zero vectors.

(i) If $\mathbf{v}_i \cdot \mathbf{v}_j \leq 0$ for all $i \neq j$, then $m \leq 2n$.

(ii) Let each $\mathbf{v}_i$ be a unit vector. If $\mathbf{v}_i \cdot \mathbf{v}_j \leq -\varepsilon < 0$ for all $i \neq j$, $m \leq 1 + \frac{1}{\varepsilon}$.

Proof.

(i) We shall prove this by induction on n. The base case $n = 0$ is clear as then we have $m = 0$ as well.

Since we only care about the sign of $\mathbf{v}_i \cdot \mathbf{v}_j$, assume without loss of generality that $\mathbf{v}_m = (1, 0, 0, \ldots, 0)$. For each $i \in [m-1]$, let $\mathbf{v}_i = (\alpha_i, v_{i,1}, v_{i,2}, \ldots, v_{i,m-1})$ and $\mathbf{w}_i = (v_{i,1}, v_{i,2}, \ldots, v_{i,m-1})$. Then as $\mathbf{v}_i \cdot \mathbf{v}_m \leq 0$ for each $i \in [m-1]$, we have $\alpha_i \leq 0$ for each such i.

We now claim that at most one of the $\mathbf{w}_i$s can be equal to the all zero vector $\mathbf{0}$. To prove this, assume otherwise (w.l.o.g.) that $\mathbf{w}_j = \mathbf{w}_{m-1} = \mathbf{0}$ for some j. Then

$$\mathbf{w}_j \cdot \mathbf{w}_{m-1} = \alpha_j \alpha_{m-1} > 0 \quad \text{(as each } \mathbf{v}_i \text{ is non-zero)}$$

Thus assume w.l.o.g. that $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{m-2}$ are all non-zero vectors that also have non-zero $\mathbf{w}_i$ for each i. Note that for each $i, j \in [m-2]$,

$$\mathbf{y}_i \cdot \mathbf{y}_j = \mathbf{w}_i \mathbf{w}_j - \alpha_i \alpha_j \leq \mathbf{v}_i \cdot \mathbf{v}_j \leq \mathbf{0}.$$

Applying the induction on the $\mathbf{y}_i$s for $i \in [m-2]$, we have

$$m - 2 \leq 2(n-1).$$

The result follows.

(ii) Let $\mathbf{z} = \mathbf{v}_1 + \mathbf{v}_2 + \cdots + \mathbf{v}_m$. Then

$$\begin{aligned} 0 &\leq \|\mathbf{z}\|^2 \\ &= \sum_{i=1}^{m} \|\mathbf{v}_i\|^2 + 2\sum_{i<j} \mathbf{v}_i \cdot \mathbf{v}_j \\ &\leq m + 2\binom{m}{2}(-\varepsilon) \\ &= m(1 - \varepsilon m + \varepsilon). \end{aligned}$$

The result follows.

■

§3. Introduction to Probability

§§3.1. Introduction

Definition 3.1. A *probability space* is a triple $(\Omega, \mathcal{F}, P)$, such that

(i) Ω is a non-empty set called the *sample space*.

(ii) $\mathcal{F}$ is a subset of 2^Ω (the power set of Ω) called the *event space*, such that

- $\Omega \in \mathcal{F}$,
- if $A \in \mathcal{F}$, then $\Omega \setminus A \in \mathcal{F}$, and
- $\mathcal{F}$ is closed under countable unions. That is, if $A_1, A_2, \ldots \in \mathcal{F}$ then $\bigcup_{i \in \mathbb{N}} A_i \in \mathcal{F}$.

(iii) P, the *probability distribution*, is a function from $\mathcal{F}$ to $[0, 1]$ such that

- $P(\Omega) = 1$ and
- if $A_1, A_2, \ldots \in \mathcal{F}$ is a collection of pairwise disjoint sets, then

$$P\left(\bigcup_{i \in X} A_i\right) = \sum_{i \in X} P(A_i).$$

We shall restrict ourselves to the case where Ω is a finite set.
We abuse notation and for $\omega \in \Omega$, denote $P(\{\omega\})$ as $P(\omega)$.

Definition 3.2. Let $\mathbb{D}$ be a finite set. The *uniform distribution over* $\mathbb{D}$, denoted $\mathscr{U}_\mathbb{D}$, is the one corresponding to the probability space $(\mathbb{D}, 2^\mathbb{D}, p)$, where

$$p(A) = \frac{|A|}{|\mathbb{D}|} \text{ for any } A \subseteq \mathbb{D}.$$

Definition 3.3. Let $(\Omega, \mathcal{F}, P)$ be a probability space. A (real-valued) *random variable* is a function $X : \Omega \to \mathbb{R}$ such that

$$\{\omega \in \Omega \mid X(\omega) \leq r\} \in \mathcal{F} \text{ for all } r \in \mathbb{R}.$$

In the above case, the *expectation* of X is defined as

$$\mathbb{E}[X] = \sum_{\omega \in \Omega} P(\omega) X(\omega).$$

In this report, we primarily consider binary random variables, that is, random variables which map to $\{0, 1\}$.

Definition 3.4. Let $(\Omega, \mathcal{F}, P)$ be a probability space. Given an *event* $E \in \mathcal{F}$, we define its *indicator variable* to be the random variable $\mathbb{1}_E : \Omega \to \{0, 1\}$ such that for each $\omega \in \Omega$,

$$\mathbb{1}_E(\omega) = \begin{cases} 1 & \text{if } \omega \in E \\ 0 & \text{otherwise.} \end{cases}$$

We occasionally abuse notation and use E instead of $\mathbb{1}_E$.

Now that we have the concept of a random variable, we can talk of the probability that the random variable has a given value. For example, given a probability space $(\Omega, \mathcal{F}, P)$, a corresponding random variable V, and $x \in \mathbb{R}$, we can write

$$\Pr[V \geq x] = P\left(\{\omega \in \Omega \mid V(\omega) \geq x\}\right).$$

The right expression is well-defined due to the property given in 3.3.

And now that we have the above, we can *abstract away* the details of Ω and $\mathcal{F}$. We can talk merely of the different elements of the image of the random variable and the associated probabilities.

Definition 3.5. The *Bernoulli distribution* with parameter $p \in [0,1]$ is the binary random variable X such that for $x \in \{0,1\}$

$$\Pr[X = x] = \begin{cases} p & \text{if } x = 1 \\ 1-p & \text{if } x = 0. \end{cases}$$

Definition 3.6. The *binomial distribution* with parameters $n \in \mathbb{N}$ and $p \in [0,1]$ is the random variable B to $\{0,1,2,\ldots,n\}$ such that for $k \in \{0,1,2,\ldots,n\}$,

$$\Pr[B = k] = \binom{n}{k} p^k (1-p)^{n-k}.$$

§§3.2. Some Results

Lemma 3.1. Let $(\Omega, \mathcal{F}, P)$ be a probability space and $E \in \mathcal{F}$ be any event. Then

$$\mathbb{E}[\mathbb{1}_E] = P(E)$$

Theorem 3.2 (Linearity of Expectation)**.** Given random variables $V_1, V_2, \ldots, V_m$ defined over the same domain $\mathbb{D}$ and with the same probability distribution p,

$$\mathbb{E}\left[\sum_{i=1}^{m} V_i\right] = \sum_{i=1}^{m} \mathbb{E}[V_i].$$

Theorem 3.3. Let X be a binomial distribution with parameters n and p. Then

$$\mathbb{E}[X] = np.$$

Proof. We have

$$\begin{aligned}
\mathbb{E}[X] &= \sum_{i=0}^{n} i \Pr[X = i] \\
&= \sum_{i=1}^{n} i \binom{n}{i} p^i (1-p)^{n-i} \\
&= \sum_{i=1}^{n} n \binom{n-1}{i-1} p^i (1-p)^{n-i} \\
&= \sum_{i=1}^{n} np \binom{n-1}{i-1} p^{i-1} (1-p)^{n-i} \\
&= np \sum_{i=0}^{n-1} \binom{n-1}{i} p^i (1-p)^{n-1-i} \\
&= np(p + 1 - p)^{n-1} \\
&= np.
\end{aligned}$$

■

Theorem 3.4 (Union Bound)**.** Let $(\Omega, \mathcal{F}, P)$ be a probability space. Given events $E_1, E_2, \ldots, E_m$,

$$P\left(\bigcup_{i=1}^{m} E_i\right) \leq \sum_{i=1}^{m} \Pr(E_i).$$

The union bound is tight if for every $i, j \in [m]$ such that $i \neq j$,

$$E_i \cap E_j = \varnothing$$

We omit the proofs of the above results as they are relatively easy to check.

Theorem 3.5 (Markov Bound)**.** Let V be a non-negative random variable. Then for any $t > 0$,

$$\Pr[V \geq t] \leq \frac{\mathbb{E}[V]}{t}.$$

Proof. We have

$$\begin{aligned}
\mathbb{E}[V] &= \sum_{i \in [0,t)} i \Pr[V = i] + \sum_{i \in [t,\infty)} i \Pr[V = i] \\
&\geq \sum_{i \in [t,\infty)} i \Pr[V = i] \\
&\geq t \sum_{i \in [t,\infty)} \Pr[V = i] \\
&= t \Pr[V \geq t].
\end{aligned}$$

■

Corollary 3.6. Let V be a non-negative random variable. Then for any $a \geq 1$,

$$\Pr[V \geq a\mathbb{E}[V]] \leq \frac{1}{a}.$$

Putting $t = a\mathbb{E}[V]$ in the Markov bound gives the required result.

Definition 3.7. Two random variables A and B are called *independent* if for every a, b in the ranges of A, B respectively,

$$\Pr[(A = a) \vee (B = b)] = \Pr[A = a] \Pr[B = b].$$

Definition 3.8. Let X, Y be two random variables defined over the same probability space. Let X take the distinct values $x_1, x_2, \ldots, x_n$ and Y take the distinct values $y_1, y_2, \ldots, y_m$. For some i, j, we then define the probability of $X = x_i$ *conditioned* over $Y = y_j$ as

$$\Pr[X = x_i \mid Y = y_j] = \frac{\Pr[X = x_i \wedge Y = y_j]}{\Pr[Y = y_j]}$$

The above is straightforward to check using the definition of conditional probability.

Theorem 3.7 (Multiplicative Chernoff Bound)**.** Let $X_1, X_2, \ldots, X_m$ be independent binary random variables and $X = \sum X_i$. Then for $0 < \varepsilon \leq 1$,

$$\Pr[|X - \mathbb{E}[X]| > \varepsilon \mathbb{E}[X]] < 2e^{-\varepsilon^2 \mathbb{E}[X]/3}$$

Theorem 3.8 (Additive Chernoff Bound)**.** Let $X_1, X_2, \ldots, X_m$ be independent binary random variables and $X = \sum X_i$. Then for $0 < \varepsilon \leq 1$,

$$\Pr[|X - \mathbb{E}[X]| > \varepsilon m] < 2e^{-\varepsilon^2 m/2}$$

The Chernoff bounds can be proved by applying the Markov bound 3.5 to e^{tX} to get

$$\mathbb{E}[X \geq a] \leq \min_{t>0} e^{-ta} \prod_i \mathbb{E}[e^{tX_i}]$$

and

$$\mathbb{E}[X \leq a] \leq \min_{t>0} e^{ta} \prod_i \mathbb{E}[e^{-tX_i}]$$

and bounding the resultant expression after putting a suitable value of a.

§§3.3. The Probabilistic Method

The probabilistic method is a method used to show the existence of objects that exhibit certain properties without giving an explicit construction.
Say we must show the existence of an object $\mathscr{C}$ that has property $\mathscr{P}$. This is done by defining a probability distribution $\mathscr{D}$ over all such objects and showing that when an object $\mathscr{C}$ is chosen according to $\mathscr{D}$,

$$\Pr[\mathscr{C} \text{ has property } \mathscr{P}] > 0 \text{ or } \Pr[\mathscr{C} \text{ doesn't have property } \mathscr{P}] < 1$$

This can be simplified by defining sub-properties $P_1, P_2, \ldots, P_m$ such that $\mathscr{P} = P_1 \wedge P_2 \wedge \cdots \wedge P_m$ and then showing that for all valid i,

$$\Pr[\mathscr{C} \text{ doesn't have property } P_i] < \frac{1}{m}$$

and then using the union bound 3.4.

Finally, if f is a function from the set of objects to $\mathbb{R}$, then $\mathbb{E}[f(\mathscr{C})] \leq b$ for some $b \in \mathbb{R}$ implies that there exists an object $\mathscr{C}_0$ such that $f(\mathscr{C}_0) \leq b$.

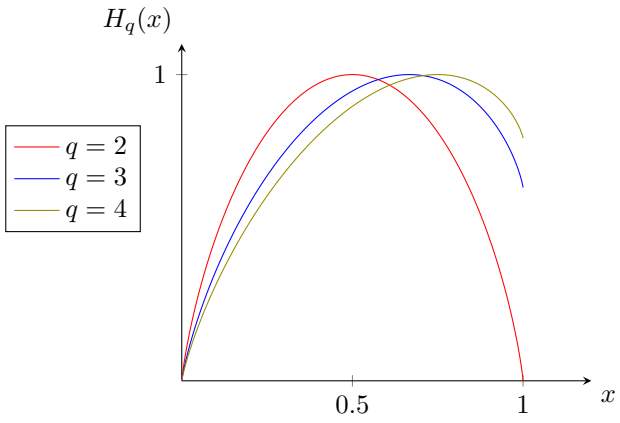
§§3.4. The Entropy Function

Definition 3.9. Let $q \in \mathbb{Z}$ and $x \in \mathbb{R}$ such that $q \geq 2$ and $0 \leq x \leq 1$. Then the *q-ary entropy function* is defined as follows:

$$H_q(x) = x \log_q(q-1) - x \log_q(x) - (1-x) \log_q(1-x).$$

We take $0 \log_q(0) = 0$.

The following graph shows the behaviour of $H_q(x)$ for some values of q.



H_q attains its maximum value of 1 at $1 - \frac{1}{q}$.

The binary entropy function H_2 is denoted as H_{Ber}.

Note that

$$q^{-H_q(p)} = \left(\frac{p}{q-1}\right)^p (1-p)^{1-p}$$

§4. Introduction

§§4.1. Why is Coding Theory required?

The English language has an enormous amount of redundancy. For instance, it's likely that the reader has seen the following text or something like it:

`Fi yuo cna raed tihs yuo hvae a sgtrane mnid.  Olny srmat poelpe cna raed tish.`

While we have no doubt about the smartness of the reader, the above is *not* a good test of the same. It merely means that the massive amount of redundancy in the language allows effective communication even in the presence of (an acceptable amount of) errors.
Of course, even in the digital realm, we expect to see errors as no system is truly foolproof. To understand the data even in the presence of errors, digital systems use redundancy as well.

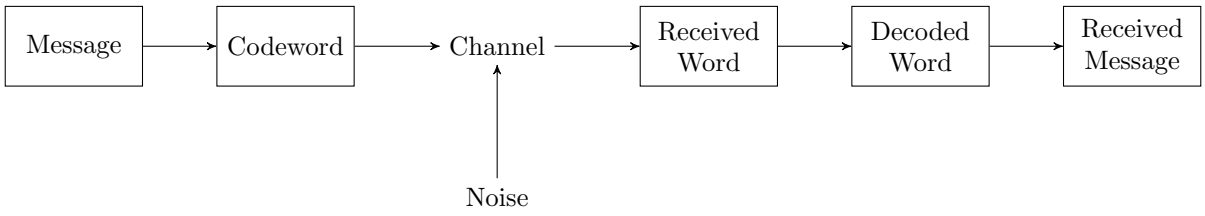
Error-correcting codes (or just codes) are clever ways of representing data by introducing some redundancy such that the original data we want transmitted can be recovered even if parts of the data have errors.
When packets are transmitted over the internet, some packets get corrupted or lost in transmission. To deal with data corruption, a form of correction called "CRC Checksum" is used. This is not a very good code. It searches for errors, and if an error is detected, it requests the data again. However, for obvious reasons, this is not always feasible. For instance, if we are receiving a transmission from a Mars Rover, we cannot just request the information again, it is simply not practical. Codes can also be seen in non-communication examples such as bank balances, bar codes and the memory of a computer. In these cases as well', the data cannot be requested again.

In this report, we shall mainly focus on codes in the communication scenario. There is a sender who wants to send symbols over a noisy channel. He first encodes the symbols into a *codeword* of n symbols and sends it over the *channel*. The receiver gets a *received word* of n symbols. He then tries to *decode* the received word to recover the original symbols.

We make the assumption in this text that the sender and receiver have no method to communicate outside of the channel.

As we mentioned earlier, redundancy enables us to detect errors in a code with higher likelihood. A basic question that comes to mind is "What is the minimum amount of redundancy required to ensure a high probability of detecting all errors in a code?"

The following diagram shows essentially what occurs in the process of encoding and decoding.



During the course of this report, we primarily follow the texts *A First Course in Coding Theory* [2] for sections 4 through 6 and *Essential Coding Theory* [1] for sections 7 through 9.

§§4.2. Basics and Definitions

Definition 4.1. A *block code* C over an alphabet Σ is a non-empty subset of Σ^n for some $n \in \mathbb{N}$.

Henceforth, we shall refer to "block code" as just "code".
A *q-ary code of length n* is a subset of Σ^n where $|\Sigma| = q$.

Definition 4.2. Elements of a code are called *codewords*. The *length* of a code C over an alphabet Σ is the n for which $C \subseteq \Sigma^n$.

A code C of cardinality M and length n can be written as an $M \times n$ array whose rows are the codewords of C.

Definition 4.3. Let C be a code of cardinality M over Σ where $|\Sigma| = q$. Then the *dimension* of C is given by

$$k = \log_q(M)$$

Example. Let us look at two codes over $\mathbb{F}_2 = \{0, 1\}$. The first code is called the *parity code*, denoted $C_\oplus$. Given any $(x_1, x_2, x_3) \in \{0, 1\}^3$, its corresponding codeword is

$$C_\oplus((x_1, x_2, x_3)) = (x_1, x_2, x_3, x_1 + x_2 + x_3).$$

That is, the final bit gives the sum of the first three bits modulo 2. If a single error (a single bit-flip) occurs in $C_\oplus$, we can *detect* it, since then the sum of the first three bits modulo 2 will not be equal to the final bit. The second, called the *repetition code* (represented $C_{n,\text{rep}}$), which is a very naïve approach involves repeating each bit n times. For instance, for $n = 3$, we have

$$C_{3,\text{rep}}((x_1, x_2, x_3)) = (x_1, x_1, x_1, x_2, x_2, x_2, x_3, x_3, x_3)$$

$C_{3,\text{rep}}$ is stronger since if a bit-flip occurs, not only can we detect it, we can *correct* it and recover the original message by taking the symbol repeated 2 or more times in each set of 3 bits.

We shall now attempt to formalize the meanings of encoding and decoding. As we wish to send a message through a channel by converting it to a codeword and then sending the codeword, we may use $[|C|]$ to list all the messages that we can send.

Definition 4.4. Let $C \subseteq \Sigma^n$. An equivalent description of the code C is an injective mapping $E : [|C|] \to \Sigma^n$ called the *encoding function*.

To decode on the other hand, we must obtain a message from whatever word we receive (which may have errors).

Definition 4.5. Let $C \subseteq \Sigma^n$ be a code. A mapping $D : \Sigma^n \to [|C|]$ is called a *decoding function* of C.

Definition 4.6. For $\mathbf{x} = (x_1, x_2, \ldots, x_n), \mathbf{y} = (y_1, y_2, \ldots, y_n) \in \Sigma^n$, we define the *Hamming distance* between $\mathbf{x}$ and $\mathbf{y}$ as

$$d(\mathbf{x}, \mathbf{y}) = |\{i \in [n] \mid x_i \neq y_i\}|.$$

If Σ is a field, then for $\mathbf{x} \in \Sigma^n$, we define the *Hamming weight* of $\mathbf{x}$ to be

$$\text{wt}(\mathbf{x}) = d(\mathbf{x}, \mathbf{0})$$

where $\mathbf{0}$ represents the all zero vector $(0, 0, \ldots, 0)$.

We see that the Hamming distance d defines a metric on Σ^n as

1. $d(\mathbf{x}, \mathbf{y}) = 0$ if and only if $\{i \in [n] \mid x_i \neq y_i\} = \varnothing$. This is equivalent to saying that $x_i = y_i$ for all $i \in [n]$, that is, $\mathbf{x} = \mathbf{y}$.
2. $d(\mathbf{x}, \mathbf{y}) = |\{i \in [n] \mid x_i \neq y_i\}| = |\{i \in [n] \mid y_i \neq x_i\}| = d(\mathbf{y}, \mathbf{x})$.
3. Note that the minimum number of steps required to change $\mathbf{x}$ to $\mathbf{z}$ is $d(\mathbf{x}, \mathbf{z})$. We can change $\mathbf{x}$ to $\mathbf{z}$ by changing $\mathbf{x}$ to $\mathbf{y}$ in $d(\mathbf{x}, \mathbf{y})$ steps then $\mathbf{y}$ to $\mathbf{z}$ in $d(\mathbf{y}, \mathbf{z})$ steps. This gives $d(\mathbf{x}, \mathbf{z}) \leq d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{z})$.

Although the Hamming distance metric may not always be a very appropriate metric, it provides a good way to measure how "close" two strings are.

We assume the following noise model called the *Adversarial Noise Model*, which was first studied by Hamming:
Any error pattern can occur during transmission as long as the total number of errors is bounded. This means that both the location and the nature of the errors is arbitrary.

We define error correction and detection in terms of the (Hamming) distance between a codeword and the word that is received after passing the codeword through the channel. Note that the output word is not fixed for a given codeword since we assume the Adversarial Noise Model.

Definition 4.7. Let $C \subseteq \Sigma^n$ and let $t \geq 1$ be an integer. C is said to be *t-error correcting* if there exists a decoding function D such that for every $\mathbf{m} \in [|C|]$, $\mathbf{y} \in \Sigma^n$ where

(i) $d(C(\mathbf{m}), \mathbf{y}) \leq t$ and

(ii) $C(\mathbf{m})$ can become $\mathbf{y}$ after passing through the channel,

we have $D(\mathbf{y}) = \mathbf{m}$.

Definition 4.8. Let $C \subseteq \Sigma^n$ and let $t \geq 1$ be an integer. C is said to be *t-error detecting* if there exists a detecting procedure $D : \Sigma^n \to \{0, 1\}$ such that for every $\mathbf{m} \in [|C|]$, $\mathbf{y} \in \Sigma^n$ where

1. $1 \leq d(C(\mathbf{m}), \mathbf{y}) \leq t$ and
2. $C(\mathbf{m})$ can become $\mathbf{y}$ after passing through the channel,

we have $D(\mathbf{y}) = 1$ if $\mathbf{y} \in C$ and 0 otherwise.

So going back to the example discussed, $C_{\oplus}$ is a 1-error detecting code and $C_{3,\text{rep}}$ is a 1-error correcting code (and a 2-error detecting code).

Definition 4.9. If we receive a word $\mathbf{y}$ after passing a codeword through a channel, *nearest neighbour decoding* or *minimum distance decoding* decodes $\mathbf{y}$ as codeword $\mathbf{x}'$ such that $d(\mathbf{x}', \mathbf{y})$ is minimum.

Definition 4.10. If we receive a word $\mathbf{y}$ after passing a codeword through a channel, *maximum likelihood decoding* decodes $\mathbf{y}$ as codeword $\mathbf{x}'$ such that $\Pr(\mathbf{y} \text{ received} \mid \mathbf{x} \text{ sent})$ is maximum.

We now consider a specific type of channel.

Definition 4.11. Consider an alphabet Σ. A corresponding channel is called a *q-ary symmetric channel* if

(i) Each symbol has the same probability $p < \frac{1}{2}$, called the *symbol error probability*, of becoming erroneous.

(ii) If a symbol becomes erroneous, then each of the $q-1$ other symbols of Σ is equally likely to replace it.

A q-ary symmetric channel is denoted $q\mathsf{SC}_p$ and a binary symmetric channel is denoted BSC_p.
Note that if the error vector $\mathbf{e}$ is drawn from $q\mathsf{SC}_p$, $\text{wt}(\mathbf{e})$ follows a binomial distribution with parameters n and p.

The probability that a received codeword of length n has an error in exactly i specific places is $p^i(1-p)^{n-i}$. Since $p < \frac{1}{2}$, it is more probable that a fewer number of errors occur.

Consider the code $C = \{000, 111\}$ for the binary alphabet $\{0, 1\}$ passed through a binary symmetric channel. Say 000 is transmitted. Then following nearest neighbour decoding, the probability that the received codeword is decoded as 000 (that is, the received codeword is 000, 100, 010 or 001) is $(1-p)^3 + 3p(1-p)^2 = (1-p)^2(1+2p)$. For any word $\mathbf{c}$ in C, the *word error probability* of C, denoted $P_{\text{err}}(\mathbf{c})$, denotes the probability that $\mathbf{c}$ is interpreted incorrectly after passing through a channel (Note that by symmetry, this is equal for any codeword in this case). Here,

$$P_{\text{err}}(\mathbf{c}) = 1 - (1-p)^2(1+2p) = 3p^2 - 2p^3.$$

Definition 4.12. For any code C, the *minimum distance* is defined as

$$d(C) = \min\{d(\mathbf{x}, \mathbf{y}) \mid \mathbf{x}, \mathbf{y} \in C, \mathbf{x} \neq \mathbf{y}\}.$$

Theorem 4.1.

(i) A code C is s-error detecting if $d(C) > s$.

(ii) A code C is s-error correcting $d(C) > 2s$.

Proof.

(i) Suppose a codeword $\mathbf{x}$ is transmitted, and the received codeword $\mathbf{y}$ has s or fewer errors. Then $d(\mathbf{x}, \mathbf{y}) \leq s$ and as $\mathbf{x} \in C$, $\mathbf{y} \notin C$ and the error can be detected.

(ii) Supposed a codeword $\mathbf{x}$ is transmitted and the received codeword $\mathbf{y}$ has s or fewer errors. Then $d(\mathbf{x}, \mathbf{y}) \leq s$. We claim that the codeword $\mathbf{x}'$ such that $d(\mathbf{x}', \mathbf{y})$ is minimum is unique and equal to $\mathbf{x}$. If $\mathbf{x}' \neq \mathbf{x}$, then $d(\mathbf{y}, \mathbf{x}') \leq d(\mathbf{x}, \mathbf{y}) \leq s$. Then $d(\mathbf{x}, \mathbf{x}') \leq d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{x}') \leq s + s = 2s$ as d is a metric. However, we have $d(C) > 2s$, which is a contradiction. Thus, $\mathbf{x}' = \mathbf{x}$ and C can correct up to s errors.

■

Corollary 4.2. Let a code C have minimum distance d. Then

(i) C is $(d-1)$-error detecting.

(ii) C is $\left\lfloor \frac{d-1}{2} \right\rfloor$-error correcting.

Proof. We have $d > s$ if and only if $s \leq d - 1$ and $d > 2s$ if and only if $s \leq \left\lfloor \frac{d-1}{2} \right\rfloor$. Combining this with 4.1 gives the required result. ■

Notation. An $(n, M, d)_q$*-code* is a code of length n, cardinality M and minimum distance d over an alphabet Σ such that $|\Sigma| = q$.
For example, the code $\{000, 111\}$ over $\{0, 1\}$ is a $(3, 2, 3)_2$-code.

Definition 4.13. Let C be a code of length n and minimum distance d. The *relative distance* of C is given by $\delta = \frac{d}{n}$.

Definition 4.14. For $q, n \in \mathbb{N}$, the *repetition code of length* n over an alphabet Σ is the code whose codewords are $aa \cdots a$ (repeated n times) where $a \in \Sigma$.

A q-ary repetition code of length n is an $(n, q, n)_q$-code. It is represented by $C_{n,\text{rep}}$.

§5. Bounds on the Number of Codewords

An ideal $(n, M, d)_q$-code has a small value of n (so the data can be transmitted faster), a large value of M (so a larger number of messages can be transmitted), and a large value of d (to detect/correct many errors).

§§5.1. Some Useful Bounds

We shall attempt to optimize the value of M keeping the other two fixed.

Definition 5.1. We denote by $A_q(n, d)$ the largest value of M such that there exists a q-ary $(n, M, d)_q$-code.

Theorem 5.1. For all $n, q \in \mathbb{N}$,

(i) $A_q(n, 1) = q^n$.

(ii) $A_q(n, n) = q$.

Proof.

(i) $M \leq q^n$ as the code is a subset of Σ^n. As Σ^n is of length n and has minimum distance 1, $A_q(n, 1) = q^n$.

(ii) If the minimum distance of a code is n, any two codewords differ at all n places. Thus the symbols appearing in any fixed position in the M codewords must be distinct, giving $M \leq q$. The q-ary repetition code of length n is an $(n, q, n)_q$-code so $A_q(n, n) = q$.

■

Definition 5.2. Two q-ary codes are *equivalent* if one can be obtained from the other by a combination of the following operations:

(i) permutation of the positions of the code.

(ii) permutation of the symbols appearing in a fixed position.

Note that if we represent an $(n, M, d)_q$-code as an $M \times n$ array, (i) corresponds to rearranging the columns and (ii) corresponds to a renaming of the symbols in a given column.
As distances between codewords remain the same, two equivalent codes have the same value of length, cardinality, and minimum distance.

Lemma 5.2. Any $(n, M, d)_q$-code over $\{0, 1, \ldots, q-1\}$ is equivalent to an $(n, M, d)_q$-code (over $\{0, 1, \ldots, q-1\}$) that contains the codeword $\mathbf{0} = 000 \cdots 0$.

Proof. Consider any codeword $x = x_1 x_2 \cdots x_n$. For each i, apply the permutation of symbols

$$\begin{pmatrix} x_i & 0 & j & \\ \downarrow & \downarrow & \downarrow & \text{for all } j \neq x_i, 0 \\ 0 & x_i & j & \end{pmatrix}.$$

■

Let $\Sigma = \mathbb{F}_2$. For some $n \in \mathbb{N}$, consider x, y be two vectors in Σ^n, where $x = x_1 x_2 \cdots x_n$ and $y = y_1 y_2 \cdots y_n$. $x + y$ is given by the component wise sum in $\mathbb{F}_2$. $x \cap y$ is defined to be the component wise multiplication in $\mathbb{F}_2$. That is, $x + y = (x_1 + y_1, x_2 + y_2, \ldots, x_n + y_n)$ and $x \cap y = (x_1 y_1, x_2 y_2, \ldots, x_n y_n)$.

Lemma 5.3. For any $x, y \in \mathbb{F}_2^n$, $d(x, y) = \mathrm{wt}(x + y)$.

Proof. $x + y$ has a 1 wherever $x_i \neq y_i$ and 0 elsewhere. As $\mathrm{wt}(x)$ is just the number of 1s in x, $d(x, y) = \mathrm{wt}(x + y)$. ■

Lemma 5.4. For any $x, y \in \mathbb{F}_2^n$, $d(x, y) = \mathrm{wt}(x) + \mathrm{wt}(y) - 2\,\mathrm{wt}(x \cap y)$.

Proof. We have $d(x,y) = \text{wt}(x+y)$, the number of 1s in $x+y$, from the previous lemma. This is given by the number of positions with 1 in x + the number of positions with 1 in y − the number of positions with 1 in both x and y. This is just $\text{wt}(x) + \text{wt}(y) - 2\,\text{wt}(x \cap y)$. ■

Theorem 5.5. Let d be odd. Then a binary $(n, M, d)_2$-code exists if and only if a binary $(n+1, M, d+1)_2$-code exists.

Proof. Let C be a binary $(n, M, d)_2$-code for odd d. Construct a code C' such that for each $x = x_1x_2\cdots x_n \in C$, we extend it to $x' = x_1x_2\cdots x_{n+1}$ where $x_{n+1} = \sum_{i=1}^{n} x_i$. This is called adding an *overall parity check*. As $\text{wt}(x')$ is even for any $x' \in C'$ (because $\text{wt}(x') \equiv 2x_{n+1} \pmod 2$), $d(C')$ is even by 5.4. Clearly, $d \le d(C') \le d+1$. As d is odd, $d(C') = d+1$. Thus C' is an $(n+1, M, d+1)_2$ code.

In the other direction, let C' be a binary $(n+1, M, d+1)_2$-code. Choose $x, y \in C'$ such that $d(x,y) = d+1$. Construct a code by choosing any position where they differ and deleting this position from all codewords. The resulting code has minimum distance d and is thus an $(n, M, d)_2$-code. ■

Corollary 5.6. Let d be odd. then $A_2(n+1, d+1) = A_2(n, d)$.

Proof. This follows from the previous theorem. ■

Definition 5.3. For any $u \in \Sigma^n$ (where $|\Sigma| = q$) and any integer $r \ge 0$, the *Hamming ball* or *sphere* of radius r and centre u is given by

$$B_q(u, r) = \{v \in \Sigma^n \mid d(u,v) \le r\}.$$

If q is understood, we simply write $B(u, r)$.

This gives more insight into why a code is s-error correcting if $d(C) > 2s$ (if u is received, the transmitted codeword will be the unique codeword in $B_q(u, s)$).

Lemma 5.7. Let Σ contain q symbols. The number of words in a ball of radius r in Σ^n is exactly

$$\sum_{i=0}^{r} \binom{n}{i} (q-1)^i.$$

Proof. The number of words at a distance of exactly i from a given word x is given by choosing exactly i positions from the n positions and then picking one of $q-1$ symbols to replace the symbol at each of those positions. This is equal to $\binom{n}{i}(q-1)^i$. The required result is the sum of this quantity over $\{i : 0 \le i \le r\}$. ■

Theorem 5.8 (The Hamming bound)**.** An $(n, M, 2t+1)_q$-code satisfies

$$M \sum_{i=0}^{t} \binom{n}{i} (q-1)^i \le q^n.$$

Proof. Note that two balls of radius t centered at distinct codewords have no words in common as the minimum distance of the code is $2t+1$. The number of words in each ball is $\sum_{i=0}^{t} \binom{n}{i}(q-1)^i$ by 5.7. The total number of words in the M balls is M multiplied by this quantity, which must be less than or equal to q^n, the total number of words in Σ^n. This gives the required result. ■

The Hamming bound provides an upper bound on $A_q(n, d)$.
In general for an $(n, M, d)_q$-code,

$$M \sum_{i=0}^{\lfloor \frac{d-1}{2} \rfloor} \binom{n}{i} (q-1)^i \le q^n.$$

In the binary case, the Hamming bound gives

$$M \sum_{i=0}^{\lfloor \frac{d-1}{2} \rfloor} \binom{n}{i} \le 2^n.$$

For an $(n, M, d)_q$-code of dimension k,

$$k \leq n - \log_q \left(\sum_{i=0}^{\lfloor \frac{d-1}{2} \rfloor} \binom{n}{i} (q-1)^i \right)$$

§§5.2. Perfect Codes

A code which achieves the Hamming bound is called a *perfect code*.

Definition 5.4. An $(n, M, d)_q$-code is called a *perfect code* if it satisfies

$$M \sum_{i=0}^{\lfloor \frac{d-1}{2} \rfloor} \binom{n}{i} (q-1)^i = q^n.$$

The repetition code of length n, where n is odd, is a perfect code. These codes, along with codes that contain exactly one codeword and codes that are the entirety of Σ^n, are called *trivial* perfect codes.

An example of a nontrivial perfect code is the following.
Let $\mathbf{0} = 0000000$, $\mathbf{1} = 1111111$, $a_1 = 1101000$, $a_2 = 0110100$, $a_3 = 0011010$, $a_4 = 0001101$, $a_5 = 1000110$, $a_6 = 0100011$, $a_7 = 1010001$. We further define b_i as the same as a_i except that all 0s are replaced with 1s and all 1s are replaced with 0s. Then the code C containing $\mathbf{0}, \mathbf{1}$, all the a_is and all the b_is is a (nontrivial) perfect code. Note that the a_is correspond to the rows of the matrix in 1.2.

For any $i, j \in [7]$, $i \neq j$,

$$d(a_i, a_j) = \operatorname{wt}(a_i) + \operatorname{wt}(a_j) - 2\operatorname{wt}(a_i + a_j) = 3 + 3 - 2 = 4.$$

Also, $d(\mathbf{0}, a_i) = d(\mathbf{1}, b_i) = 3$ and $d(\mathbf{0}, b_i) = d(\mathbf{1}, a_i) = 4$.
Finally, a_i and b_j differ exactly where a_i and a_j agree so $d(a_i, b_j) = 7 - d(a_i, a_j) = 3$.
Thus C is a $(7, 16, 3)_2$-code. It may be checked that this is a perfect code.

We shall study perfect codes more in detail later on.

Theorem 5.9. If there exists a Hadamard $(4t-1, 2t-1, t-1)$-design, then

$$A_q(4t-1, 2t-1) \geq 8t.$$

Proof. Similar to the construction described above, construct a code C containing $\mathbf{0}$, $\mathbf{1}$, the vectors a_i corresponding to each of the rows of the incidence matrix of the Hadamard design and the vectors b_i which are the same as a_i except that all 0s are replaced with 1s and all 1s are replaced with 0s.
As each vertex is present in exactly $2t-1$ blocks, there are $(2t-1)$ 1s in each row and thus

$$d(\mathbf{0}, a_i) = 2t - 1 \text{ for all valid } i.$$

Similarly,

$$d(\mathbf{0}, b_i) = d(\mathbf{1}, a_i) = 2t \text{ and } d(\mathbf{1}, b_i) = 2t - 1.$$

We also have

$$\begin{aligned} d(a_i, a_j) &= \operatorname{wt}(a_i) + \operatorname{wt}(a_j) - 2\operatorname{wt}(a_i \cap a_j) \\ &= 2t - 1 + 2t - 1 - 2\operatorname{wt}(a_i \cap a_j). \end{aligned}$$

As mentioned earlier, $\operatorname{wt}(a_i) = \operatorname{wt}(a_j) = 2t - 1$. $\operatorname{wt}(a_i \cap a_j)$ is the number of blocks in which the vertices corresponding to a_i and a_j are both present, which is equal to $t - 1$.
Thus, $d(a_i, a_j) = (2t-1) + (2t-1) - 2(t-1) = 2t - 1$ for all valid i, j. Similarly, we get $d(b_i, b_j) = 2t - 1$ and $d(a_i, b_j) = 2t$.
The number of codewords in this code is $2 + 2(4t-1) = 8t$. The minimum distance of this code is $2t - 1$ and its length is $4t - 1$.
Therefore, the resulting code is a $(4t-1, 8t, 2t-1)_2$-code. ∎

This provides another bound on $A_q(n, d)$. Namely, if a Hadamard $(4t - 1, 2t - 1, t - 1)$-design exists, $A_q(4t - 1, 2t - 1) \geq 8t$.

§6. Linear Codes

§§6.1. Introduction to Linear Codes

Definition 6.1. A *linear code* over $\mathbb{F}_q$ is a subspace of $V(n,q)$ for some positive integer n.

Note that the dimension (recall 4.3) of a linear code is equal to its dimension as a vector space.
We denote by $\mathbf{0}$ the element of a linear code that consists of all 0s.
If a linear code is a k-dimensional subspace of $V(n,q)$, we call it an $[n,k]_q$-code. If we further wish to specify the minimum distance d of the code, we call it an $[n,k,d]_q$-code.

Lemma 6.1. If $x, y \in V(n,q)$, then $d(x,y) = \mathrm{wt}(x-y)$.

Proof. $x-y$ is nonzero exactly wherever x and y differ. The result follows. ■

Theorem 6.2. Let C be a linear code and $\mathrm{wt}(C) = \min\{\mathrm{wt}(x) \mid x \in C \setminus \{0\}\}$. Then $d(C) = \mathrm{wt}(C)$.

Proof. There exist $x, y \in C$ such that $d(x,y) = d(C)$, that is, $\mathrm{wt}(x-y) = d(C)$. This gives $d(C) = \mathrm{wt}(x-y) \geq \mathrm{wt}(C)$ since $x - y \in C$.
Let $x \in C$ such that $\mathrm{wt}(x) = \mathrm{wt}(C)$. Then $\mathrm{wt}(C) = d(x, \mathbf{0}) \geq d(C)$.
This gives $d(C) = \mathrm{wt}(C)$. ■

Note that to find the minimum distance in any general code, we must make $\binom{m}{2}$ comparisons, but in a linear code, we only need to examine the weights of $M-1$ codewords.

Definition 6.2. Let C be an $[n,k]_q$-code. A $k \times n$ matrix whose rows form a basis of C is called a *generator matrix* of C.

Note that if G is a generator matrix of an $[n,k]_q$-code C, $C = \{\mathbf{x}G \mid \mathbf{x} \in V(n,k)\}$.
For example, the generator matrix of the q-ary repetition code of length n over $\mathbb{F}_q$ is the $1 \times n$ matrix $\begin{pmatrix} 1 & 1 & \cdots & 1 \end{pmatrix}$.

Definition 6.3. Two linear codes over $\mathbb{F}_q$ are called *equivalent* if one can be obtained from the other by a combination of the following operations:

(i) Permutation of the positions of the code.

(ii) Multiplication of the symbols appearing in a fixed position by a non-zero scalar.

Alternatively, two linear codes are equivalent if they are isomorphic.
Note that this is *not* the same as the definition of equivalence we gave earlier in 5.2.

Theorem 6.3. Two $k \times n$ matrices generate the same $[n,k]_q$-code over $\mathbb{F}_q$ if one matrix can be obtained from the other by a combination of the following operations:

(i) Permutation of the rows.

(ii) Multiplication of a row by a non-zero scalar.

(iii) Addition of scalar multiple of one row to another.

(iv) Permutation of the columns.

(v) Multiplication of a column by a non-zero scalar.

Proof. The first three conditions merely replace one basis of the code with another. The final two conditions are those in the definition of equivalence of linear codes. ■

Theorem 6.4. Let G be a generator matrix of an $[n,k]_q$-code. G generates the same code as as a matrix in the *standard form* $(I_k \mid A)$, where I_k is the $k \times k$ identity matrix and A is a $k \times (n-k)$ matrix.

Proof. Let $G = (g_{ij})$ and let $\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_k$ and $\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_n$ be the rows and columns of the matrix respectively. We repeat the following three step procedure for $j = 1, 2, \ldots, k$, which transforms $\mathbf{c}_j$ into the required form leaving the first $j-1$ columns unchanged. Suppose that G has already been transformed to

$$\begin{pmatrix} 1 & 0 & \cdots & 0 & g_{1j} & \cdots & g_{1n} \\ 0 & 1 & \cdots & 0 & g_{2j} & \cdots & g_{2n} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 & g_{j-1,j} & \cdots & g_{j-1,n} \\ 0 & 0 & \cdots & 0 & g_{jj} & \cdots & g_{jn} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 0 & g_{nj} & \cdots & g_{nn} \end{pmatrix}.$$

1. if $g_{jj} = 0$ and $g_{ij} \neq 0$ for some $i > j$, we interchange $\mathbf{r}_i$ and $\mathbf{r}_j$. Otherwise, if $g_{jj} = 0$ and $g_{ji} \neq 0$ for some $i > j$, interchange $\mathbf{c}_i$ and $\mathbf{c}_j$ (the existence of such an i is guaranteed by the fact that the rows are linearly independent).
2. Multiply $\mathbf{r}_j$ with g_{jj}^{-1} (which is well-defined as $g_{jj} \neq 0$).
3. For each $i = 1, 2, \ldots, k$, replace $\mathbf{r}_i$ with $\mathbf{r}_i - g_{ij}\mathbf{r}_j$.

The column $\mathbf{c}_j$ then has the required form. After we repeat this procedure for $j = 1, 2, \ldots, k$, the generator matrix will be in standard form. ■

§§6.2. Encoding and Decoding with Linear Codes

Let C be an $[n, k]_q$-code with generator matrix G over $\mathbb{F}_q$. For any $\mathbf{u} \in V(k, q)$ (here we represent $\mathbf{u}$ by a row vector), we have $\mathbf{u}G \in C$ as this is merely a linear combination of the row vectors of G.

This suggests a way to encode message vectors of $\mathbb{F}_q^k$. Note that the encoding function briefly described above given by $\mathbf{u} \mapsto \mathbf{u}G$ for $\mathbf{u} \in V(k, q)$ maps the vector space $V(k, q)$ onto C.

This is even easier to understand in the case where the generator matrix is in standard form. Let $G = (I_k \mid A)$ where $A = (a_{ij}i$ implies is a $k \times (n-k)$ matrix. The message vector $\mathbf{u}$ is encoded as $\mathbf{x} = \mathbf{u}G = x_1 x_2 \cdots x_k x_{k+1} \cdots x_n$. Here, $x_i = u_i$ for $1 \leq i \leq k$ and $x_i = \sum_{j=1}^{k} a_{ji} u_j$ for $k+1 \leq i \leq n$.
Note that in addition to the message $\mathbf{u}$, $\mathbf{x}$ contains extra information. The message digits $x_{k+1}, x_{k+2}, \ldots, x_n$ are called *check digits* and represent the redundancy we mentioned at the start of this report. They provide protection against any errors that might occur.

Now, suppose the codeword $\mathbf{x}$ is sent through the channel and the received codeword is $\mathbf{y}$. We define the *error vector* $\mathbf{e}$ to be

$$\mathbf{e} = \mathbf{y} - \mathbf{x}.$$

Definition 6.4. Suppose that C is an $[n, k]_q$-code over $\mathbb{F}_q$ and $\mathbf{a} \in V(n, q)$. Then for $\mathbf{a} \in C$, the *coset* $\mathbf{a} + C$ is given by

$$\mathbf{a} + C = \{\mathbf{a} + \mathbf{x} \mid \mathbf{x} \in C\}.$$

This corresponds to 1.11 considering $V(n, q)$ as a group under addition.

Lemma 6.5. The set of cosets of a code in $V(n, q)$ partition $V(n, q)$. Furthermore, for $\mathbf{a}, \mathbf{b} \in V(n, q)$, $\mathbf{a} + C = \mathbf{b} + C$ if and only if $\mathbf{b} \in \mathbf{a} + C$.

Proof. This follows from 1.10. ■

Theorem 6.6 (Lagrange's Theorem)**.** Suppose C is an $[n, k]_q$-code over $\mathbb{F}_q$. Then every coset of C in $V(n, q)$ contains exactly q^k elements.

Proof. This follows from 1.11. ■

Definition 6.5. A vector having minimum weight in a coset is called a *coset leader*. If a coset has more than one vector of minimum weight, we choose any such vector and call it the coset leader.

A *Slepian standard array* or simply *standard array* of an $[n,k]_q$-code C is a $q^{n-k} \times q^k$ array of the elements of $V(n,q)$ which is constructed as follows.

1. List the codewords of C, starting with $\mathbf{0}$, in the first row.
2. Choose any vector of minimum weight not already in the array. Write this as the first entry of the following row. Denote this vector as the coset leader.
3. Fill out the row by adding the coset leader to the codeword at the top of each column. The sum of the coset leader of row i and the jth codeword becomes the i,jth element of the array.
4. Repeat the above two steps until all the cosets and every vector of $V(n,q)$ appears in the array.

That is, each row of the standard array represents a coset with the coset leader written on the left.
Note that any element of the array is equal to the sum of the first element of the row and column of said element.
For example, the standard array corresponding to the code $C = \{0000, 1011, 0101, 1110\}$ is

0000	1011	0101	1110
1000	0011	1101	0110
0100	1111	0001	1010
0010	1001	0111	1100

Now, if we want to decode a received vector, we may do so by identifying the error vector $\mathbf{e}$ with the first element of the row containing the received vector and the decoded word as the first element of the column. That is, we decode an element as the codeword at the top of its column in the standard array.

The error vectors which will be corrected are precisely the coset leaders.

Let C be an $[n,k,2t+1]_q$-code. Then C can correct any t errors. This implies every vector of weight $i \leq t$ is a coset leader. Determining the number of coset leaders of weight $i > t$ is problematic in the general case however. It is easy to establish that in the case of perfect codes, this is equal to 0 for each $i > t$. However, these values are not known even for several well-known families of codes.

The primary issues with standard array decoding are that

- It requires a massive amount of storage as we store every single vector in $V(n,q)$. For example, a binary code of length 32 would require 2^{32} entries.
- It takes a large amount of time to locate a given vector in the array due to its size.

§§6.3. Some results on Binary Linear Codes

We now restrict ourselves to binary linear codes. We assume that the channel is a binary symmetric channel with symbol error probability p.

Theorem 6.7. Let C be a binary $[n,k]_2$-code and for $i = 0, 1, \ldots, n$ let α_i denote the number of coset leaders of weight i. Then the probability that a decoded vector decoded using a standard array is the codeword $\mathbf{c}$ which was sent is

$$P_{\text{corr}}(\mathbf{c}) = \sum_{i=0}^{n} \alpha_i p^i (1-p)^{n-i}$$

Proof. The probability that the error vector is a given vector of weight i is $p^i(1-p)^{n-i}$. As there are α_i such errors, the probability that the error vector is one of the acceptable error vectors is the sum of $\alpha_i p^i (1-p)^{n-i}$ over $i = 0, 1, 2, \ldots, n$. ■

The probability that the decoded word is *not* the codeword $\mathbf{c}$ sent, called the *word error rate*, is given by

$$P_{\text{err}}(\mathbf{c}) = 1 - P_{\text{corr}}(\mathbf{c}).$$

Definition 6.6. Let C be an $[n,k]_q$-code. The *rate* of C is defined by

$$R(C) = \frac{k}{n}.$$

The rate captures a way to measure the redundancy of a code. The higher the redundancy, the lower the rate is. Therefore, an efficient code will have a high rate.

A natural question to ask would be:
Given a code of distance d, what is the largest rate R that it can have?

§§6.4. Error Detection in Binary Linear Codes

We now consider error *detection*. If the codeword sent is $\mathbf{x}$, we will fail to detect an error if and only if the received vector $\mathbf{y}$ is a codeword as well, that is, $\mathbf{e} = \mathbf{y} - \mathbf{x}$ is a codeword.
For any code C and $\mathbf{c} \in C$, we denote by $P_{\text{undetec}}(\mathbf{c})$ the probability that an incorrect codeword is received, which is independent of the codeword sent in the binary symmetric channel case.

Theorem 6.8. Let C be a binary $[n,k]_2$-code transmitted through a binary symmetric channel of symbol error probability p. Let A_i be the number of codewords of C of weight i for each valid i. Then if C is used for error detection, for any $\mathbf{c} \in C$,

$$P_{\text{undetec}}(\mathbf{c}) = \sum_{i=1}^{n} A_i p^i (1-p)^{n-i}.$$

Proof. We must simply find the probability that the error vector is in C. As the probability that there are exactly i specific errors is $p^i(1-p)^{n-i}$ and there are A_i codewords of weight i, the result follows. ■

If we detect an error, we might ask to retransmit the data again. In this case, the probability that we will request retransmission is given by

$$P_{\text{retrans}}(\mathbf{c}) = 1 - (1-p)^n - P_{\text{undetec}}(\mathbf{c}).$$

The above follows as $(1-p)^n$ is the probability that no error occurs and P_{undetec} is the probability that an error occurs but we do not detect it.

§§6.5. The Dual Code

Definition 6.7. Let C be a linear $[n,k]_q$-code. The *dual code* of C, denoted $C^\perp$, is the orthogonal subspace of C with respect to $V(n,q)$, that is,

$$C^\perp = \{\mathbf{v} \in V(n,q) \mid \mathbf{v} \perp \mathbf{w} \text{ for all } \mathbf{w} \in C\}.$$

Lemma 6.9. Let C be an $[n,k]_q$-code with generator matrix G. Then $\mathbf{v} \in V(n,q)$ is an element of $C^\perp$ if and only if $\mathbf{v}G^{\mathrm{T}} = O$.

Proof. Let $\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_k$ be the row vectors of G which form a basis of C.
If $\mathbf{v} \in C^\perp$, then $\mathbf{v}$ is orthogonal to every element of C and in particular, the row vectors of G^{T}, so the 'only if' part of the lemma follows.
To prove the 'if' part of the lemma, let $\mathbf{u}$ be any element of C. We have that $\mathbf{v} \cdot \mathbf{r}_i$ for each i as $\mathbf{v}G^{\mathrm{T}} = 0$. Then $u = a_1\mathbf{r}_1 + a_2\mathbf{r}_2 + \cdots + a_k\mathbf{r}_k$ for scalars $a_1, a_2, \ldots, a_k$ and so

$$\begin{aligned}\mathbf{v} \cdot \mathbf{u} &= \mathbf{v} \cdot (a_1\mathbf{r}_1 + a_2\mathbf{r}_2 + \cdots + a_k\mathbf{r}_k) \\ &= a_1(\mathbf{v} \cdot \mathbf{r}_1) + a_2(\mathbf{v} \cdot \mathbf{r}_2) + \cdots + a_k(\mathbf{v} \cdot \mathbf{r}_k) = 0\end{aligned}$$

This proves the required result. ■

Theorem 6.10. Let C be a linear $[n,k]_q$-code over $\mathbb{F}_q$. Then $C^\perp$ is a linear $[n,n-k]_q$-code.

Proof. Let us first show that $C^\perp$ is a linear code. Let $\mathbf{v}_1, \mathbf{v}_2 \in C^\perp$. Then for all $\alpha, \beta \in \mathbb{F}_q$ and $\mathbf{u} \in C$, $(\alpha\mathbf{v}_1 + \beta\mathbf{v}_2)\cdot\mathbf{u} = \alpha(\mathbf{v}_1\cdot\mathbf{u}) + \beta(\mathbf{v}_2\cdot\mathbf{u}) = 0$. That is, $\alpha\mathbf{v}_1 + \beta\mathbf{v}_2 \in C^\perp$. Thus $C^\perp$ is a subspace of $V(n,q)$ and is a linear code.
We shall now show that $C^\perp$ has dimension $n-k$. Let $G = (g_{ij})$ be a generator matrix of C. Then $C^\perp$ contains exactly those vectors $\mathbf{v} = v_1v_2\cdots v_k$ satisfying

$$\sum_{i=1}^{n} g_{ij}v_j = 0 \text{ for } i = 1,2,\ldots,k.$$

It is a standard result that the solution space of a system of k independent homogeneous equations and n unknowns has dimension $n-k$. Thus $C^\perp$ is of dimension $n-k$. ∎

§§6.6. The Parity-Check Matrix

Definition 6.8. Let C be a linear $[n,k]_q$-code. A *parity-check matrix* H of C is a generator matrix of $C^\perp$.

Thus H is an $(n-k)\times n$ matrix that satisfies $GH^{\mathrm{T}} = O$, where G is a generator matrix of C.
Note that 6.9 gives

$$C = \{\mathbf{x} \in V(n,q) \mid \mathbf{x}H^{\mathrm{T}} = O\}.$$

The rows of a parity-check matrix give parity checks on the corresponding code. That is, they say that certain linear combinations of the coordinates are equal to 0.
For example, if a code C has parity-check matrix

$$H = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$$

then the code is given by

$$C = \{(x_1,x_2,x_3,x_4) \in V(4,2) \mid x_1 + x_2 = x_3 + x_4 = 0\}.$$

Lemma 6.11. Let G be a generator matrix of an $[n,k]_q$-code C_1. If H is an $(n-k)\times n$ parity check matrix of a code C_2 such that $GH^{\mathrm{T}} = O$, then $C_1 = C_2$.

Proof. We shall first show $C_1 \subseteq C_2$. Given any $\mathbf{u} \in C_1$, there exists $\mathbf{x} \in V(n,q)$ such that $\mathbf{u} = \mathbf{x}G$. Then

$$\mathbf{u}H^{\mathrm{T}} = (\mathbf{x}G)H^{\mathrm{T}} = \mathbf{x}(GH^{\mathrm{T}}) = 0.$$

That is, $\mathbf{u} \in C_2$.
To prove the converse, note that $\dim C_2 = n - (n-k) = \dim C_1$. As $C_1 \subseteq C_2$ and $\dim C_1 = \dim C_2$, $C_1 = C_2$. ∎

Theorem 6.12. If $G = (I_k \mid A)$ is the standard form generator matrix of a linear $[n,k]_q$-code C, then a parity check matrix of C is $H = (-A^{\mathrm{T}} \mid I_{n-k})$.

Proof. H is an $(n-k)\times n$ matrix so it is of the correct size. We shall show that every row of G is orthogonal to every row of H. Let

$$G = \left(\begin{array}{ccc|ccc} 1 & \cdots & 0 & a_{11} & \cdots & a_{1,n-k} \\ \vdots & & \vdots & \vdots & & \vdots \\ 0 & \cdots & 1 & a_{n1} & \cdots & a_{n,n-k} \end{array}\right).$$

Then

$$H = \left(\begin{array}{ccc|ccc} -a_{11} & \cdots & -a_{n1} & 1 & \cdots & 0 \\ \vdots & & \vdots & \vdots & & \vdots \\ -a_{1,n-k} & \cdots & -a_{n,n-k} & 0 & \cdots & 1 \end{array}\right).$$

Then the inner product of the ith row of G and the jth row of H is $0+\cdots+0+(-a_{ij})+0+\cdots+0+(a_{ij})+0+\cdots+0 = 0$ ∎

Definition 6.9. An $(n-k) \times n$ parity check matrix H is said to be in *standard form* if $H = (B \mid I_{n-k})$.

Theorem 6.13. Let C be an $[n, k, d]_q$-code with parity-check matrix H. Then d is the minimum number of linearly dependent columns in H.

Proof. By 6.2, we must show that d, the minimum weight of a nonzero codeword in C is equal to t, the minimum number of linearly dependent columns.
Let $\mathbf{c} = (c_1, c_2, \ldots, c_n) \in C$ such that $\mathrm{wt}(\mathbf{c}) = d$. We have $\mathbf{c}H^{\mathrm{T}} = O$, which gives that

$$\sum_{i=1}^{n} c_i H_i = \mathbf{0}$$

where H_i represents the ith column of H. Note that we can skip multiplication for the terms where $c_i = 0$. This leaves $\mathrm{wt}(c)$ linearly dependent columns. Thus $d \geq t$.
For the other direction, let $H_{i_1}, H_{i_2}, \ldots, H_{i_t}$ be linearly dependent. Then there exist nonzero scalars $c'_{i_1}, \ldots, c'_{i_t}$ such that

$$\sum_{j=1}^{t} c'_{i_j} H_{i_j} = \mathbf{0}.$$

The c_{i_j}s are nonzero due to the minimality of t. Now let $\mathbf{c'} = (c'_1, c'_2, \ldots, c'_n)$ where $c'_j = 0$ for $j \notin \{i_1, i_2, \ldots, i_t\}$. This gives $\mathbf{c'}H^{\mathrm{T}} = O$ and thus $c' \in C$. This implies $d = \mathrm{wt}(c') \leq t$. The required result follows. ■

§§6.7. Syndrome Decoding

Definition 6.10. Let H be a parity-check matrix of an $[n, k]_q$-code C. Then for any vector $\mathbf{y} \in V(n, q)$, the $1 \times (n-k)$ row vector

$$\mathrm{S}(\mathbf{y}) = \mathbf{y}H^{\mathrm{T}}$$

is called the *syndrome* of $\mathbf{y}$.

Note the following.

- If the rows of H are $\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_{n-k}$, then for $\mathbf{y} \in C$,

$$\mathrm{S}(\mathbf{y}) = (\mathbf{y} \cdot \mathbf{h}_1, \mathbf{y} \cdot \mathbf{h}_2, \ldots, \mathbf{y} \cdot \mathbf{h}_{n-k}).$$

- $S(\mathbf{y}) = 0 \iff \mathbf{y} \in C$.

Lemma 6.14. Let C be a linear code and $\mathbf{u}, \mathbf{v} \in C$. $\mathbf{u}$ and $\mathbf{v}$ are in the same coset of C if and only if they have the same syndrome.

Proof. $\mathbf{u}$ and $\mathbf{v}$ are in the same coset

$$\begin{aligned}
&\iff \mathbf{u} - \mathbf{v} \in C \\
&\iff (\mathbf{u} - \mathbf{v})H^{\mathrm{T}} = 0 \\
&\iff \mathbf{u}H^{\mathrm{T}} = \mathbf{v}H^{\mathrm{T}} \\
&\iff \mathrm{S}(\mathbf{u}) = \mathrm{S}(\mathbf{v})
\end{aligned}$$

■

Corollary 6.15. There is a bijection between cosets and syndromes.

In standard array decoding, one of the issues that we faced was that the time taken to locate a vector is very large. We fix this by calculating the syndrome $\mathrm{S}(\mathbf{e})$ for each coset leader $\mathbf{e}$ and extend the standard array by listing the syndromes in an extra column.

For example, consider the code $C = \{0000, 1011, 0101, 1110\}$. A parity-check matrix of C is given by

$$H = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}$$

Thus, the modified standard array corresponding to the code $C = \{0000, 1011, 0101, 1110\}$ is

0000	1011	0101	1110	00
1000	0011	1101	0110	11
0100	1111	0001	1010	01
0010	1001	0111	1100	10

The decoding algorithm is as follows.

1. When a vector $\mathbf{y}$ is received, calculate $\mathrm{S}(\mathbf{y}) = \mathbf{y}H^{\mathrm{T}}$ and locate $\mathrm{S}(\mathbf{y})$ in the syndromes column of the array.
2. Locate $\mathbf{y}$ in the corresponding row and decode it as the codeword at the top of the column containing $\mathbf{y}$.

This works because if $\mathbf{y} = \mathbf{x} + \mathbf{e}$, where $\mathbf{x}$ is the codeword sent and $\mathbf{e}$ is the error vector,

$$\mathrm{S}(\mathbf{y}) = (\mathbf{x} + \mathbf{e})H^{\mathrm{T}} = \mathbf{e}H^{\mathrm{T}} = \mathrm{S}(\mathbf{e}).$$

The second issue we had in standard array decoding was that we had to store all the elements of $V(n, q)$ in the array. However, note that now we only need to store the syndromes and the coset leaders (and the code, of course) in the computer memory. This is called a *syndrome look-up table*.

§7. Perfect Codes

§§7.1. Binary Hamming Codes

Definition 7.1. Define the $r \times (2^r - 1)$ matrix $\mathbf{H}_r$ over $\mathbb{F}_2$, such that the ith column of $\mathbf{H}_r$, $1 \leq i \leq 2^r - 1$ is the binary representation of i.

For example,

$$\mathbf{H}_3 = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}.$$

Definition 7.2. For $r > 1$, the $[2^r - 1, 2^r - r - 1]_2$-code that has parity-check matrix $\mathbf{H}_r$ is called the *Hamming code* and is denoted $C_{H,r}$.

In other words, the code $C_{H,r}$ is given by

$$C_{H,r} = \{\mathbf{c} \in \{0,1\}^{2^r-1} \mid \mathbf{c}\mathbf{H}_r^{\mathrm{T}} = O\}.$$

Theorem 7.1. For $r > 1$, the $[2^r - 1, 2^r - r - 1]_2$ Hamming code has minimum distance 3.

Proof. Due to 6.13, this is equivalent to showing that the minimum number of linearly dependent columns in $\mathbf{H}_r$ is 3. Since distinct numbers have distinct binary representations, the sum of two columns cannot be equal to 0 so the minimum distance is ≥ 3. It is equal to 3 as the sum of the first three columns of $\mathbf{H}_r$ is 0.

$$\begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

■

Theorem 7.2. For $r > 1$, the code $C_{H,r}$ is a perfect code.

Proof. The code $C_{H,r}$ is a $[2^r - 1, 2^r - r - 1, 3]_2$-code. It may be checked that this satisfies the condition for a perfect code. ■

Decoding Hamming codes using syndrome decoding is very effective due to the nature of the code.

1. When a vector $\mathbf{y}$ is received, calculate its syndrome $\mathrm{S}(\mathbf{y}) = \mathbf{y}H^{\mathrm{T}}$.
2. If $\mathrm{S}(\mathbf{y}) = 0$, then assume that $\mathbf{y}$ was the codeword sent.
3. If $\mathrm{S}(\mathbf{y}) \neq 0$, then assuming a single error, $\mathrm{S}(\mathbf{y})$ gives the binary representation of the error position and so the error can be corrected.

This works because the syndrome of $00\cdots010\cdots00$ (with 1 in the jth position) is simply the transpose of the jth column of H, which is the binary representation of j.

For example, if we consider $\mathbf{H}_3$ and $\mathbf{y} = 1101011$, then $\mathrm{S}(\mathbf{y}) = 110$, indicating that the error is in the 6th position and $\mathbf{y}$ must be decoded as 1101001.

We now generalize the Hamming code.

Definition 7.3. Define the $r \times n$ matrix $H_{q,r}$ where each column is a nonzero vector from $V(r,q)$ such that the first nonzero entry is 1.

For example,

$$H_{3,2} = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 2 \end{pmatrix}$$

Definition 7.4. For $r > 1$, the $\left[\frac{q^r-1}{q-1}, \frac{q^r-1}{q-1} - r\right]_q$ -code which has generator matrix equal to $H_{q,r}$ is called the *q-ary Hamming code* and is denoted $C_{H,r,q}$.

Theorem 7.3. $C_{H,n,q}$ has minimum distance 3.

Proof. As no two columns are linearly dependent, the minimum distance of $C_{H,n,q}$ must be ≥ 3. It is equal to 3 as

$$\begin{pmatrix}0\\0\\1\end{pmatrix} + \begin{pmatrix}0\\1\\0\end{pmatrix} + \begin{pmatrix}0\\1\\1\end{pmatrix} = \begin{pmatrix}0\\0\\0\end{pmatrix}.$$

■

Theorem 7.4. $C_{H,n,q}$ is a perfect code.

Proof. It may be checked that the parameters of the q-ary Hamming code satisfy the Hamming bound. ■

Thus, $C_{H,n,q}$ is a single error-correcting code.

Corollary 7.5. If q is a prime power and $n = \frac{q^r-1}{q-1}$ for some integer $r > 1$, then

$$A_q(n, 3) = q^{n-r}$$

Proof. $C_{H,n,q}$ is a perfect $(n, M, 3)$-code where $n = \frac{q^r-1}{q-1}$ and $M = q^{n-r}$. ■

Decoding q-ary Hamming codes is also done using syndrome decoding.

1. When a vector $\mathbf{y}$ is received, calculate its syndrome $\mathrm{S}(\mathbf{y}) = \mathbf{y}H^{\mathrm{T}}$.
2. If $\mathrm{S}(\mathbf{y}) = 0$, then assume that $\mathbf{y}$ was the codeword sent.
3. If $\mathrm{S}(\mathbf{y}) \neq 0$, then assuming a single error, $\mathrm{S}(\mathbf{y}) = b\mathbf{H}_j^{\mathrm{T}}$ for some $b \in \mathbb{F}_q$ and j where $\mathbf{H}_j$ represents the jth column of H. The error is corrected by subtracting b from the jth entry of y.

§§7.2. Family of Codes

Definition 7.5. Let $q \geq 2$. Let $(n_i)_{i\geq 1}$ be an increasing sequence (of lengths) and there exist sequences $(M_i)_{i\geq 1}$ and $(d_i)_{i\geq 1}$ such that for each $i \geq 1$, there exists an (n_i, M_i, d_i)-code C_i. We also define k_i to be the dimension of C_i for each i. Then the sequence $(C_i)_{i\geq 1}$ is said to be a *family of codes*.

Definition 7.6. Let $C = (C_i)_{i\geq 1}$ be a family of codes and let k_i be the dimension of C_i for each $i \geq 1$. The *rate* of C is defined by

$$R(C) = \lim_{i\to\infty} \left(\frac{k_i}{n_i}\right).$$

The *relative distance* of C is defined by

$$\delta(C) = \lim_{i\to\infty} \left(\frac{d_i}{n_i}\right).$$

For example, consider C_H, the family of binary Hamming codes with $n_i = 2^i - 1$, $k_i = 2^i - i - 1$ and $d_i = 3$. Then

$$R(C_H) = \lim_{i\to\infty} \left(1 - \frac{i}{2^i - 1}\right) = 1$$

and

$$\delta(C_H) = \lim_{i\to\infty} \left(\frac{3}{2^i - 1}\right) = 0.$$

Earlier, we mentioned that we desire codes that have high rates. Or more precisely, given the minimum distance d, what is the largest rate that the code can have? However, this comparison is slightly unfair since we are comparing an raw parameter with a ratio of two parameters. Now, we desire families of codes that have both high rates *and* high relative distances. The following question, which makes more sense than the previous one, is the one we will now attempt to answer:
What is the optimal tradeoff between $R(C)$ and $\delta(C)$ for a given family of codes C?

§§7.3. The Hadamard and Simplex codes

Definition 7.7. For $r > 1$, the *Simplex code*, denoted $C_{\text{Sim},r}$ is given by $C_{H,r}^{\perp}$.

Note that this is merely the code which has generator matrix equal to $\mathbf{H}_r$.

Definition 7.8. For $r > 1$, the *Hadamard code*, denoted $C_{\text{Had},r}$ is the code which has generator matrix equal to the resultant matrix on adding an all 0s column to $\mathbf{H}_r$.

Both the Simplex code and the Hadamard code are $[2^r - 1, r]_2$-codes.

Theorem 7.6. For $r > 1$, $C_{\text{Had},r}$ is a $[2^r - 1, r, 2^{r-1}]$-code.

Proof. We shall in fact show that every non-zero codeword in $C_{\text{Had},r}$ has weight 2^{r-1} and the result will follow from 6.2.
For any codeword $\mathbf{c}$, we have $\mathbf{c} = \mathbf{x}\mathbf{H}_r^{\mathrm{T}}$ for some nonzero $\mathbf{x} = (x_1, x_2, \ldots, x_r)$ in $V(r, q)$. As $\mathbf{x}$ is nonzero, assume that $x_i = 1$ for some i.
Note that the jth bit of $\mathbf{c}$ is $\mathbf{x} \cdot \mathbf{H}_r^j$, where $\mathbf{H}_r^j$ represents the jth row vector of $\mathbf{H}_r$.
Now, split the columns of the generator matrix $\mathbf{H}_r$ into 2^{r-1} disjoint pairs $(\mathbf{u}, \mathbf{v})$ such that $\mathbf{v} = \mathbf{u} + \mathbf{e}_i$, where $\mathbf{e}_i$ is the vector which has 1 in the ith position and 0 everywhere else. Then,

$$\mathbf{x} \cdot \mathbf{v} = \mathbf{x} \cdot \mathbf{u} + \mathbf{x} \cdot \mathbf{e}_i = \mathbf{x} \cdot \mathbf{u} + x_i = \mathbf{x} \cdot \mathbf{u} + 1.$$

That is, exactly one of $\mathbf{x} \cdot \mathbf{v}$ and $\mathbf{x} \cdot \mathbf{u}$ is 1. As the choice of the pair $(\mathbf{u}, \mathbf{v})$ was arbitrary, we have shown that for any nonzero codeword $\mathbf{c}$, $\text{wt}(\mathbf{c}) = 2^{r-1}$. ■

Theorem 7.7. For $r > 1$, $C_{\text{Sim},r}$ is a $[2^r - 1, r, 2^{r-1}]$-code.

Proof. Observe that any codeword of $C_{\text{Had},r}$ is given by padding a 0 onto the beginning of a codeword of $C_{\text{Sim},r}$. As all codewords of $C_{\text{Had},r}$ have weight 2^{r-1}, any codeword of $C_{\text{Sim},r}$ also has weight 2^{r-1}. The result follows. ■

§8. Several Bounds

In this section we describe numerous useful bounds.

§§8.1. Bounding Volume using the Entropy Function

Definition 8.1. Let $q \geq 2$ and $n \geq r \geq 1$ be integers. Then the *volume* of a Hamming ball of radius r is given by

$$\mathrm{Vol}_q(r,n) = |B_q(\mathbf{0},r)| = \sum_{i=0}^{r} \binom{n}{i}(q-1)^i.$$

Theorem 8.1. Let $q \geq 2$ be an integer and $0 \leq p \leq 1 - \frac{1}{q}$ be a real. Then

(i) $\mathrm{Vol}_q(pn,n) \leq q^{H_q(p)n}$.

(ii) for large enough n, $\mathrm{Vol}_q(pn,n) \geq q^{H_q(p)n - o(n)}$

Proof.

(i) We have

$$\begin{aligned}
1 &= \sum_{i=1}^{n} \binom{n}{i} p^i (1-p)^{n-i} \\
&\geq \sum_{i=1}^{pn} \binom{n}{i} p^i (1-p)^{n-i} \\
&= \sum_{i=1}^{pn} \binom{n}{i} (q-1)^i \left(\frac{p}{(1-p)(q-1)}\right)^i (1-p)^n \\
&\geq \sum_{i=1}^{pn} \binom{n}{i} (q-1)^i (1-p)^n \left(\frac{p}{(1-p)(q-1)}\right)^{pn} && \frac{p}{(1-p)(q-1)} \leq 1 \text{ as } p \leq 1 - \frac{1}{q} \\
&= \sum_{i=1}^{pn} \binom{n}{i} (q-1)^i (1-p)^n \left(\frac{p}{(1-p)(q-1)}\right)^{pn} \\
&= \sum_{i=1}^{pn} \binom{n}{i} (q-1)^i (1-p)^{n(1-p)} \left(\frac{p}{q-1}\right)^{pn} \\
&= \sum_{i=1}^{pn} \binom{n}{i} (q-1)^i q^{-H_q(p)n} \\
&\geq \mathrm{Vol}_q(pn,n) q^{-H_q(p)n}.
\end{aligned}$$

(i) follows.

(ii) Using Stirling's Approximation 1.7, we have

$$\begin{aligned}
\binom{n}{pn} &= \frac{n!}{(pn)!(n(1-p))!} \\
&> \frac{(n/e)^n}{(pn/e)^{pn}(n(1-p)/e)^{n(1-p)}} \cdot \frac{1}{\sqrt{2\pi p(1-p)n}} \cdot e^{\lambda_1(n) - \lambda_2(pn) - \lambda_2(n(1-p))} \\
&= \frac{1}{p^{pn}(1-p)^{n(1-p)}} l(n)
\end{aligned}$$

where

$$l(n) = \frac{e^{\lambda_1(n) - \lambda_2(pn) - \lambda_2(n(1-p))}}{\sqrt{2\pi p(1-p)n}}.$$

Note that $l(n) = q^{-o(n)}$.

Now, we have

$$\begin{aligned}\text{Vol}_q(pn, n) &= \sum_{i=1}^{pn} \binom{n}{i}(q-1)^i \\ &\geq \binom{n}{pn}(q-1)^{pn} \\ &> \frac{(q-1)^{pn}}{p^{pn}(1-p)^{n(1-p)}} l(n) \\ &= q^{H_q(p)} q^{-o(n)}.\end{aligned}$$

This proves the required result.

■

§§8.2. The Hamming Bound and the Singleton Bound

Recall the Hamming Bound 5.8 which put a bound on the dimension k in terms of n, q and d:

$$\frac{k}{n} \leq 1 - \frac{\log_q \text{Vol}_q\left(\left\lfloor \frac{d-1}{2} \right\rfloor, n\right)}{n}.$$

From 8.1, we have

$$\text{Vol}_q\left(\left\lfloor \frac{d-1}{2} \right\rfloor, n\right) \leq q^{H_q\left(\frac{\delta}{2}\right)n - o(n)}.$$

Putting everything in terms of rate and relative distance,

$$R \leq 1 - H_q\left(\frac{\delta}{2}\right) + o(1)$$

Theorem 8.2 (Singleton Bound)**.** For valid n, q, d, we have

$$A_q(n, d) \leq q^{n-d+1}.$$

Proof. Let C be an $(n, A_q(n, d), d)_q$-code. Let C' be the code of length $(n-d+1)$ code obtained by deleting the first $d-1$ letters of each codeword of C. Since the minimum distance of C is d, the words obtained after deleting the first $d-1$ letters of distinct codewords of C must also be distinct. This implies that $|C'| = |C| = A_q(n, d)$. As $|C'| \leq q^{n-d+1}$, the result follows. ■

The asymptotic version of the singleton bound gives that

$$\frac{k}{n} \leq 1 - \frac{d}{n} + \frac{1}{n}.$$

Alternatively,

$$R \leq 1 - \delta + o(1).$$

§§8.3. The Gilbert-Varshamov Bound

Theorem 8.3 (Gilbert-Varshamov Bound)**.** For valid n, q, d, we have

$$A_q(n, d) \geq \frac{q^n}{\sum_{i=1}^{d-1} \binom{n}{i}(q-1)^i}$$

Proof. Let C be a $(n, A_q(n,d), d)_q$-code. Then for all $x \in \Sigma^n$, there exists $c_x \in C$ such that $d(x, c_x) < d$. This gives

$$\left| \bigcup_{c \in C} B(c, d-1) \right| = q^n.$$

If the above equality does not hold, then there exists some $v \in \Sigma^n \setminus C$ such that $d(c,v) \geq d$ for all $c \in C$, which contradicts the maximality of C.
We now have

$$\begin{aligned} q^n &= \left| \bigcup_{c \in C} B(c, d-1) \right| \\ &\leq \sum_{c \in C} |B(c, d-1)| \\ &= A_q(n,d) |B(c, d-1)| \\ &= A_q(n,d) \operatorname{Vol}_q(d-1, n) \end{aligned}$$

Substituting the value of $\operatorname{Vol}_q(d-1,n)$ proves the required result. ■

In terms of rate and relative distance, we have $\operatorname{Vol}_q(d-1,n) \leq q^{H_q(\delta)n}$ by 8.1.

The asymptotic version of the Gilbert-Varshamov bound gives that for every $0 < \delta \leq 1 - \frac{1}{q}$ there exists a code of rate R and relative distance δ such that

$$R \geq 1 - H_q(\delta).$$

§§8.4. The Plotkin Bound

Lemma 8.4 (Mapping Lemma)**.** Let $C \subseteq [q]^n$. Then there exists a function $f : C \to \mathbb{R}^{nq}$ such that

(i) for every $\mathbf{c} \in C$, $\|f(c)\| = 1$.

(ii) for every $\mathbf{c}_1 \neq \mathbf{c}_2$ in C,

$$f(\mathbf{c}_1) \cdot f(\mathbf{c}_2) = 1 - \left(\frac{q}{q-1}\right)\left(\frac{d(\mathbf{c}_1, \mathbf{c}_2)}{n}\right)$$

Proof. Define $\varphi : [q] \to \mathbb{R}^q$ by

$$\varphi(i) = \left(\frac{1}{q}, \frac{1}{q}, \cdots, \frac{1-q}{q}, \cdots, \frac{1}{q}\right) \text{ for each } i \in [q].$$

Note that for any $i \neq j$ in $[q]$,

$$\|\varphi(i)\|^2 = \frac{q-1}{q} \text{ and } \varphi(i) \cdot \varphi(j) = -\frac{1}{q}.$$

Define the required function f as follows. For each $\mathbf{c} = (c_1, c_2, \ldots, c_n) \in [q]^n$,

$$f(\mathbf{c}) = \sqrt{\frac{q}{n(q-1)}}(\varphi(c_1), \varphi(c_2), \ldots, \varphi(c_n))$$

(Identify this vector in $(\mathbb{R}^q)^n$ to the corresponding one in R^{nq}) It may be verified by the reader that this f satisfies both conditions mentioned in the question. ■

Theorem 8.5 (Plotkin Bound)**.** Let C be an $(n, M, d)_q$-code. Then

(i) If $d = n\left(1 - \frac{1}{q}\right)$, $M \leq 2qn$.

(ii) If $d > n\left(1 - \frac{1}{q}\right)$, then $M \leq \frac{qd}{qd - (q-1)n}$.

Proof. Let $C = \{c_1, c_2, \ldots, c_M\}$. Let f be the function mentioned in the mapping lemma 8.4. For $i \neq j$ in $[M]$,

$$\begin{aligned} f(c_i) \cdot f(c_j) &= 1 - \left(\frac{q}{q-1}\right)\left(\frac{d(c_i, c_j)}{n}\right) \\ &\leq 1 - \frac{qd}{(q-1)n}. \end{aligned}$$

(i) If $d = n(1 - \frac{1}{q})$, then $f(i) \cdot f(j) \leq 0$ for all $i \neq j$, and the required result follows by the first part of 2.12.

(ii) If $d > n(1 - \frac{1}{q})$, then we have

$$f(c_i) \cdot f(c_j) \leq -\left(\frac{qd - (q-1)n}{(q-1)n}\right) \leq 0.$$

The result then follows by the second part of 2.12.

■

We now present the following bound, which is an improvement on part (i) of the Plotkin bound in the binary case.

Theorem 8.6. Let C be a binary $(n, M, \frac{n}{2})_2$-code. Then $M \leq 2n$.

Proof. Let $C = \{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_M\}$ and $\mathbf{c}_i = (c_{i,1}, c_{i,2}, \ldots, c_{i,n})$ fo each i. Consider the map $f : C \to \mathbb{R}^n$ given by

$$f(\mathbf{c}_i) = ((-1)^{c_{i,1}}, (-1)^{c_{i,2}}, \ldots, (-1)^{c_{i,n}}) \text{ for each } i.$$

For any valid $i \neq j$,

$$f(\mathbf{c}_i) \cdot f(\mathbf{c}_j) = n - 2d(\mathbf{c}_i, \mathbf{c}_j) \leq 0.$$

The result follows on using 2.12 on the $f(\mathbf{c}_i)$'s. ■

§§8.5. The Griesmer Bound

Lemma 8.7. If there exists an $[n, k, d]_q$-code, then there also exists an $[n - d, k - 1, d' \geq \lceil \frac{d}{q} \rceil]_q$-code.

Proof. Let C be an $[n, k, d]_q$-code. Let G be a generator matrix of C such that the first row vector of G is of the form $\mathbf{v} = (1, 1, \ldots, 1, 0, 0, \ldots, 0)$ where all α_is are non-zero (We may assume this by considering an equivalent code). Write G as follows.

$$G = \begin{pmatrix} 1 & \cdots & 1 & 0 & \cdots & 0 \\ * & * & * & & G' & \end{pmatrix}$$

where G' is a $(k-1) \times (n-d)$ matrix. Consider the code C' generated by G'. C' clearly has length $n - d$ and dimension $k - 1$. Let d' be the length of C'. Let $\mathbf{u} \in C'$ such that $\mathrm{wt}(\mathbf{u}) = d'$. Then there exists some $\mathbf{w} = (w_1, w_2, \ldots, w_d) \in \mathbb{F}_q^d$ such that $(\mathbf{w} \mid \mathbf{u}) \in C$, where $(\mathbf{w} \mid \mathbf{u})$ represents the concatenation of w and u. By the Pigeonhole Principle, there exists $\alpha \in \mathbb{F}_q$ such that at least $\lceil \frac{d}{q} \rceil$ of $w_1, w_2, \ldots, w_d$ are equal to α. Since $(\mathbf{w} \mid \mathbf{u}) - \alpha\mathbf{v} \in C$, we have

$$\begin{aligned} d &\leq \mathrm{wt}((\mathbf{w} \mid \mathbf{u}) - \alpha\mathbf{v}) \\ &= \mathrm{wt}((\mathbf{w} - (\alpha, \alpha, \ldots, \alpha)) \mid \mathbf{u}) \\ &= \mathrm{wt}(\mathbf{w} - (\alpha, \alpha, \ldots, \alpha)) + \mathrm{wt}(\mathbf{u}) \\ &\leq \left(d - \left\lceil \frac{d}{q} \right\rceil\right) + d' \end{aligned}$$

This gives $d' \geq \left\lceil \frac{d}{q} \right\rceil$, which proves the result. ■

Theorem 8.8 (Griesmer Bound)**.** For any $[n, k, d]_q$-code,

$$n \geq \sum_{i=0}^{k-1} \left\lceil \frac{d}{q^i} \right\rceil.$$

Proof. For a given k and d, we denote by $N_{k,d}$ the minimum value of n for which there exists an $[n, k, d]$-code. We shall prove the result by induction on k. The base case $k = 0$ is clear.
Let the result be true for $k = k_0 - 1$ and let C be an $[N_{k_0,d}, k_0, d]$-code. Then by 8.7, there exists an $[N_{k_0,d} - d, k_0 - 1, d' \geq \lceil \frac{d}{q} \rceil]$-code. By the induction, this gives

$$\begin{aligned} N_{k_0,d} - d &\geq \sum_{i=0}^{k-2} \left\lceil \frac{\lceil \frac{d}{q} \rceil}{q^i} \right\rceil \\ &\geq \sum_{i=0}^{k-2} \left\lceil \frac{d}{q^{i+1}} \right\rceil \end{aligned}$$

Thus,

$$N_{k_0,d} \geq \sum_{i=0}^{k-1} \left\lceil \frac{d}{q^i} \right\rceil$$

and the result is proved. ■

§9. Shannon's Theorem

§§9.1. Introduction and the statement of the theorem

Recall the binary symmetric channel BSC_p. We use the notation $\mathbf{e} \sim \mathsf{BSC}_p$ to denote an error vector $\mathbf{e}$ that is drawn according to the distribution induced by BSC_p.

In this section, we shall discuss Shannon's theorem which was given in his remarkable paper titled "A Mathematical Theory of Communication" that gave birth to the subject of Coding Theory (and Information Theory).

He defined a quantity called the *capacity*, which is a real number such that (reliable) communication is possible if and only if the rate is less than the capacity. That is, if the capacity is C and we desire rate $R < C$, then there exists some code of rate R that guarantees a negligible probability of incorrect communication.

Theorem 9.1 (Shannon's Theorem for BSC_p)**.** Let p, ε be reals such that $0 \leq p < \frac{1}{2}$ and $0 < \varepsilon \leq \frac{1}{2} - p$. Then the following statements are true for large enough n:

(i) There exist real $\delta > 0$, an encoding function $E : \{0,1\}^k \to \{0,1\}^n$ and a decoding function $D : \{0,1\}^n \to \{0,1\}^k$ where $k \leq \lfloor n(1 - H_{\mathsf{Ber}}(p+\varepsilon)) \rfloor$, such that for every $\mathbf{m} \in \{0,1\}^k$,

$$\Pr_{\mathbf{e} \sim \mathsf{BSC}_p}[D(E(\mathbf{m}) + \mathbf{e}) \neq \mathbf{m}] \leq 2^{-\delta n}.$$

(ii) If $k \geq \lceil n(1 - H_{\mathsf{Ber}}(p) + \varepsilon) \rceil$, then for pair of encoding function and decoding function $E : \{0,1\}^k \to \{0,1\}^n$ and $D : \{0,1\}^n \to \{0,1\}^k$ respectively, there exists $\mathbf{m} \in \{0,1\}^k$ such that

$$\Pr_{\mathbf{e} \sim \mathsf{BSC}_p}[D(E(\mathbf{m}) + \mathbf{e}) \neq \mathbf{m}] \geq \frac{1}{2}.$$

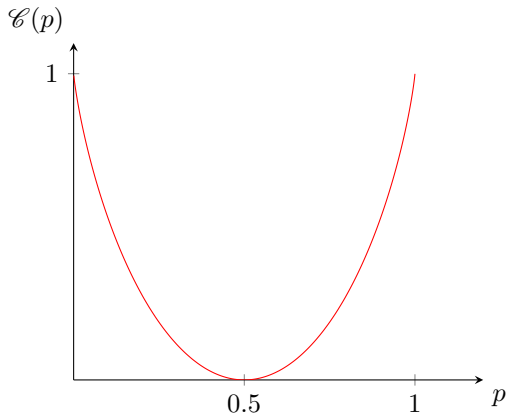
While we have only considered the binary case, a similar result holds for the q-ary case. Note that by Shannon's Theorem, the capacity of BSC_p, which we loosely defined earlier, is equal to $1 - H_{\mathsf{Ber}}(p)$. For $q\mathsf{SC}_p$, the capacity is equal to $1 - H_q(p)$.

We also state another version of Shannon's theorem as follows.

Theorem. The *capacity* $\mathscr{C}(P)$ of a binary symmetric channel of symbol error probability p is given by

$$\mathscr{C}(p) = 1 + p \log p + (1-p) \log(1-p).$$

If $0 < R < \mathscr{C}(p)$, then for any $\varepsilon > 0$, there exists for sufficiently large n, an $[n,k]_q$-code C of rate $\frac{k}{n} \geq R$ such that $P_{\text{err}}(C) < \varepsilon$.



§§9.2. Proof of the second part

If $p = 0$, we have $k \geq n(1 - H_{\mathsf{Ber}}(p) + \varepsilon) > n$ and the result follows. Therefore, we shall assume that $p > 0$. We shall prove this by contradiction. Assume that for every $\mathbf{m} \in \{0,1\}^k$, we have

$$\Pr_{\mathbf{e} \sim \mathsf{BSC}_p}[D(E(\mathbf{m}) + \mathbf{e}) \neq \mathbf{m}] < \frac{1}{2}.$$

For each $\mathbf{m}$, define

$$D_{\mathbf{m}} = \{\mathbf{y} \in \{0,1\}^n \mid D(\mathbf{y}) = \mathbf{m}\}$$

and for $\gamma > 0$, let

$$S_{\mathbf{m},\gamma} = \{\mathbf{y} \in \{0,1\}^n \mid |d(\mathbf{y}, E(\mathbf{m})) - pn| \leq \gamma pn\}.$$

Note that $S_{\mathbf{m},\gamma}$ represents the shell between radius $(1-\gamma)pn$ and $(1+\gamma)pn$ around $E(\mathbf{m})$. Now, by our assumption, we have

$$\Pr_{\mathbf{e} \sim \mathsf{BSC}_p}[E(\mathbf{m}) + \mathbf{e} \notin D_m] < \frac{1}{2}.$$

We have

$$\begin{aligned}\mathbb{E}[d(E(\mathbf{m}), \mathbf{e})] &= \mathbb{E}[d(\mathbf{0}, \mathbf{e})] \\ &= \mathbb{E}[\mathrm{wt}(\mathbf{e})] \\ &= pn \quad \text{(by 3.3)}.\end{aligned}$$

Then by the multiplicative Chernoff bound 3.7, we have

$$\begin{aligned}\Pr_{\mathbf{e} \sim \mathsf{BSC}_p}[E(\mathbf{m}) + \mathbf{e} \notin S_{\mathbf{m},\gamma}] &< 2e^{-\gamma^2 pn/3} \\ &= 2^{-\Omega(\gamma^2 n)}.\end{aligned}$$

Using the Union Bound 3.4 gives

$$\Pr_{\mathbf{e} \sim \mathsf{BSC}_p}[E(\mathbf{m}) + \mathbf{e} \notin S_{\mathbf{m},\gamma} \cap D_{\mathbf{m}}] \leq \frac{1}{2} + 2^{-\Omega(\gamma^2 n)}.$$

Then for sufficiently large n, we have

$$\Pr_{\mathbf{e} \sim \mathsf{BSC}_p}[E(\mathbf{m}) + \mathbf{e} \in D_{\mathbf{m}} \cap S_{\mathbf{m},\gamma}] \geq \frac{1}{2} - 2^{-\Omega(\gamma^2 n)} \geq \frac{1}{4}.$$

We also trivially have

$$\Pr_{\mathbf{e} \sim \mathsf{BSC}_p}[E(\mathbf{m}) + \mathbf{e} \in D_{\mathbf{m}} \cap S_{\mathbf{m},\gamma}] \leq p_{\max} \cdot |D_{\mathbf{m}} \cap S_{\mathbf{m},\gamma}|$$

where

$$\begin{aligned}p_{\max} &= \max_{\mathbf{y} \in S_{\mathbf{m},\gamma}} \Pr_{\mathbf{e} \sim \mathsf{BSC}_p}[E(\mathbf{m}) + \mathbf{e} = \mathbf{y}] \\ &\leq \max_{d \in [pn(1-\gamma), pn(1+\gamma)]} p^d (1-p)^{n-d}.\end{aligned}$$

Here the second equality arises due to the fact that the channel is BSC_p. However, since $p < \frac{1}{2}$, the function $p^d(1-p)^{n-d}$ is decreasing in d and the maximum value is attained at the minimum value of d within the range.

$$\begin{aligned}p_{\max} &\leq p^{pn(1-\gamma)}(1-p)^{n-pn(1-\gamma)} \\ &= \left(\frac{1-p}{p}\right)^{\gamma pn} p^{pn}(1-p)^{n(1-p)} \\ &= \left(\frac{1-p}{p}\right)^{\gamma pn} 2^{-nH_{\mathsf{Ber}}(p)}.\end{aligned}$$

Thus we have

$$\frac{1}{4} \leq \Pr_{\mathbf{e}\sim \mathsf{BSC}_p}[E(\mathbf{m}) + \mathbf{e} \in D_{\mathbf{m}} \cap S_{\mathbf{m},\gamma}] \leq \left(\frac{1-p}{p}\right)^{\gamma pn} 2^{-nH_{\mathsf{Ber}}(p)} \cdot |D_{\mathbf{m}} \cap S_{\mathbf{m},\gamma}|$$

which implies that

$$|D_{\mathbf{m}} \cap S_{\mathbf{m},\gamma}| \geq \frac{1}{4}\left(\frac{1-p}{p}\right)^{-\gamma pn} 2^{nH_{\mathsf{Ber}}(p)}.$$

Now note that as D is a function, the set of $D_{\mathbf{m}}$s partitions the set $\{0,1\}^n$. Thus,

$$\begin{aligned}
2^n &= \sum_{\mathbf{m}\in\{0,1\}^k} |D_{\mathbf{m}}| \\
&\geq \sum_{\mathbf{m}\in\{0,1\}^k} |D_{\mathbf{m}} \cap S_{\mathbf{m},\gamma}| \\
&\geq \sum_{\mathbf{m}\in\{0,1\}^k} \frac{1}{4}\left(\frac{1-p}{p}\right)^{-\gamma pn} 2^{nH_{\mathsf{Ber}}(p)} \\
&= 2^{k-2}\left(\frac{1}{p}-1\right)^{-\gamma pn} 2^{nH_{\mathsf{Ber}}(p)} \\
&= 2^{k-2} \cdot 2^{nH_{\mathsf{Ber}}(p) - \gamma pn \log(1/p-1)}
\end{aligned}$$

Put $\gamma = \dfrac{\varepsilon}{2p\log\left(\frac{1}{p}-1\right)}$ in the above inequality to get

$$2^n > 2^{k+nH_{\mathsf{Ber}}(p)-\varepsilon n}.$$

It follows that

$$k < n(1 - H_{\mathsf{Ber}}(p) + \varepsilon)$$

which is a contradiction and therefore the second part of the theorem is proved.

§§9.3. Proof of the first part

We prove the first part of Shannon's Theorem by the probabilistic method, the idea of which was discussed in section 3.3.
If $E(\mathbf{m})$ is the message transmitted and $\mathbf{e}$ is the error pattern, let $\mathbf{y}$ be the received word $E(\mathbf{m}) + \mathbf{e}$.
We denote by $\Pr[\mathbf{y} \mid E(\mathbf{m})]$ the probability that $\mathbf{y}$ is the received word if $E(\mathbf{m})$ is the transmitted message.
Then for any $\varepsilon' > 0$,

$$\begin{aligned}
\Pr_{\mathbf{e}\sim \mathsf{BSC}_p}[D(E(\mathbf{m}) + \mathbf{e} \neq \mathbf{m}] = &\sum_{\mathbf{y}\in B(E(\mathbf{m}),(p+\varepsilon')n)} \Pr[\mathbf{y} \mid E(\mathbf{m})] \cdot \mathbb{1}_{D(\mathbf{y})\neq\mathbf{m}} \\
&+ \sum_{\mathbf{y}\notin B(E(\mathbf{m}),(p+\varepsilon')n)} \Pr[\mathbf{y} \mid E(\mathbf{m})] \cdot \mathbb{1}_{D(\mathbf{y})\neq\mathbf{m}}
\end{aligned}$$

Simplifying the second term in the above expression,

$$\begin{aligned}
\sum_{\mathbf{y}\notin B(E(\mathbf{m}),(p+\varepsilon')n)} \Pr[\mathbf{y} \mid E(\mathbf{m})] \cdot \mathbb{1}_{D(\mathbf{y})\neq\mathbf{m}} &\leq \sum_{\mathbf{y}\notin B(E(\mathbf{m}),(p+\varepsilon')n)} \Pr[\mathbf{y} \mid E(\mathbf{m})] \\
&= \Pr[d(\mathbf{y}, E(\mathbf{m})) - pn > \varepsilon' n] \\
&\leq e^{-\varepsilon'^2 n/2} \quad \text{(by the additive Chernoff Bound 3.8)}
\end{aligned}$$

That is,

$$\Pr_{\mathbf{e}\sim\mathsf{BSC}_p}[D(E(\mathbf{m})+\mathbf{e}\neq\mathbf{m}] \leq \sum_{\mathbf{y}\in B(E(\mathbf{m}),(p+\varepsilon')n)} \Pr[\mathbf{y}\mid E(\mathbf{m})]\cdot \mathbb{1}_{D(\mathbf{y})\neq\mathbf{m}} + e^{-\varepsilon'^2 n/2}$$

We now consider a random distribution of E. For every $\mathbf{m}\in\{0,1\}^k$, pick $E(\mathbf{m})$ uniformly and independently at random from $\{0,1\}^n$. Let the decoding function D be the maximum likelihood decoding function.

Let us take the expectation on both sides of the above inequality over this distribution of E. Due to the linearity of expectation and the fact that the distributions on $\mathbf{e}$ and E are independent,

$$\mathbb{E}_E\left[\Pr_{\mathbf{e}\sim\mathsf{BSC}_p}[D(E(\mathbf{m})+\mathbf{e}\neq\mathbf{m}]\right] \leq \sum_{\mathbf{y}\in B(E(\mathbf{m}),(p+\varepsilon')n)} \Pr[\mathbf{y}\mid E(\mathbf{m})]\cdot \mathbb{E}_E\left[\mathbb{1}_{D(\mathbf{y})\neq\mathbf{m}}\right] + e^{-\varepsilon'^2 n/2}$$

We shall now simplify the right side of the above expression. By 3.4 and since D is the maximum likelihood decoding function,

$$\begin{aligned}\mathbb{E}_E\left[\mathbb{1}_{D(\mathbf{y})\neq\mathbf{m}}\right] &= \Pr_E\left[\mathbb{1}_{D(\mathbf{y})\neq\mathbf{m}}\mid E(\mathbf{m})\right]\\ &\leq \sum_{\mathbf{m}'\neq\mathbf{m}} \Pr[d(E(\mathbf{m}',\mathbf{y}))\leq d(E(\mathbf{m}),\mathbf{y})\mid E(\mathbf{m})]\end{aligned}$$

where "$\mid E(\mathbf{m})$" means that we are conditioning on the event that $E(\mathbf{m})$ is the transmitted message. As $\mathbf{y}\in B(E(\mathbf{m}),(p+\varepsilon')n)$, it follows that $d(E(\mathbf{m}),\mathbf{y})\leq(p+\varepsilon')n$. Then

$$\begin{aligned}\mathbb{E}_E\left[\mathbb{1}_{D(\mathbf{y})\neq\mathbf{m}}\right] &\leq \sum_{\mathbf{m}'\neq\mathbf{m}} \Pr[d(E(\mathbf{m}',\mathbf{y}))\leq(p+\varepsilon')n\mid E(\mathbf{m})]\\ &= \sum_{\mathbf{m}'\neq\mathbf{m}} \Pr[E(\mathbf{m}')\in B(E(\mathbf{m}),(p+\varepsilon')n)\mid E(\mathbf{m})]\\ &= \sum_{\mathbf{m}'\neq\mathbf{m}} \frac{\mathrm{Vol}_2((p+\varepsilon')n,n)}{2^n} \quad \text{(as the choice of } E(\mathbf{m})\text{) and } E(\mathbf{m}') \text{ are independent)}\\ &\leq \sum_{\mathbf{m}'\neq\mathbf{m}} 2^{n(H_{\mathsf{Ber}}(p+\varepsilon')-1)} \quad \text{(by 8.1)}\\ &< 2^k\cdot 2^{n(H_{\mathsf{Ber}}(p+\varepsilon')-1)}\\ &\leq 2^{n(1-H_{\mathsf{Ber}}(p+\varepsilon))}\cdot 2^{n(H_{\mathsf{Ber}}(p+\varepsilon')-1)} \quad \text{(due to our choice of } k)\\ &= 2^{-n(H_{\mathsf{Ber}}(p+\varepsilon)-(H_{\mathsf{Ber}}(p+\varepsilon'))}.\end{aligned}$$

Putting this back in our initial expression,

$$\mathbb{E}_E\left[\Pr_{\mathbf{e}\sim\mathsf{BSC}_p}[D(E(\mathbf{m})+\mathbf{e}\neq\mathbf{m}]\right] \leq e^{-\varepsilon'^2 n/2} + 2^{-n(H_{\mathsf{Ber}}(p+\varepsilon)-(H_{\mathsf{Ber}}(p+\varepsilon'))}\cdot \sum_{\mathbf{y}\in B(E(\mathbf{m}),(p+\varepsilon')n)} \Pr[\mathbf{y}\mid E(\mathbf{m})]$$

Then because

$$\sum_{\mathbf{y}\in B(E(\mathbf{m}),(p+\varepsilon')n)} \Pr[\mathbf{y}\mid E(\mathbf{m})] \leq \sum_{\mathbf{y}\in\{0,1\}^n} \Pr[\mathbf{y}\mid E(\mathbf{m})] = 1$$

it follows that

$$\mathbb{E}_E\left[\Pr_{\mathbf{e}\sim\mathsf{BSC}_p}[D(E(\mathbf{m})+\mathbf{e}\neq\mathbf{m}]\right] \leq e^{-\varepsilon'^2 n/2} + 2^{-n(H_{\mathsf{Ber}}(p+\varepsilon)-(H_{\mathsf{Ber}}(p+\varepsilon'))}$$

Then for large enough n and small enough δ',

$$\mathbb{E}_E\left[\Pr_{\mathbf{e}\sim\mathsf{BSC}_p}[D(E(\mathbf{m})+\mathbf{e}\neq\mathbf{m}]\right]\leq 2^{-\delta' n}$$

However, we are not yet done. We have shown that for any arbitrary $\mathbf{m}$, the expectation of the error probability is bounded above by the required quantity. However, we must show that the error probability is bounded above for all $\mathbf{m}$ *simultaneously*.
Consider the uniform random distribution of $\mathbf{m}$ over $\{0,1\}^k$. Then as the above inequality holds for all $\mathbf{m}$,

$$\mathbb{E}_{\mathbf{m}}\left[\mathbb{E}_E\left[\Pr_{\mathbf{e}\sim\mathsf{BSC}_p}[D(E(\mathbf{m})+\mathbf{e}\neq\mathbf{m}]\right]\right]\leq 2^{-\delta' n}$$

As the distributions over $\mathbf{m}$ and E are defined over different domains, we can switch the order of the expectations to get

$$\mathbb{E}_E\left[\mathbb{E}_{\mathbf{m}}\left[\Pr_{\mathbf{e}\sim\mathsf{BSC}_p}[D(E(\mathbf{m})+\mathbf{e}\neq\mathbf{m}]\right]\right]\leq 2^{-\delta' n}$$

By the probabilistic method, there exists an encoding function E^* (and decoding function D^*) such that

$$\mathbb{E}_{\mathbf{m}}\left[\Pr_{\mathbf{e}\sim\mathsf{BSC}_p}[D^*(E^*(\mathbf{m})+\mathbf{e}\neq\mathbf{m}]\right]\leq 2^{-\delta' n}$$

This says that the *average* error probability is exponentially small, while what we need to show is that the *maximum* error probability is exponentially small.
We shall show this "expurgating", which involves throwing away half the messages.

Let the messages be ordered as $\mathbf{m}_1,\mathbf{m}_2,\ldots,\mathbf{m}_{2^k}$. For each i, define

$$P_i=\Pr_{\mathbf{e}\sim\mathsf{BSC}_p}[D^*(E^*(\mathbf{m}_i)+\mathbf{e})\neq\mathbf{m}_i].$$

Assume that $P_1\leq P_2\leq\cdots\leq P_{2^k}$. We claim that $P_{2^{k-1}}\leq 2\cdot 2^{-\delta' n}$.
By the definition of P_i,

$$\begin{aligned}\frac{1}{2^k}\sum_{i=1}^{2^k}P_i&=\mathbb{E}_{\mathbf{m}}\left[\Pr_{\mathbf{e}\sim\mathsf{BSC}_p}[D^*(E^*(\mathbf{m})+\mathbf{e})\neq\mathbf{m}]\right]\\&\leq 2^{\delta' n}.\end{aligned}$$

We shall prove the claim by method of contradiction. Assume that $P_{2^{k-1}}>2\cdot 2^{-\delta' n}$. Then we have

$$\begin{aligned}\frac{1}{2^k}\sum_{i=1}^{2^k}P_i&\geq\frac{1}{2^k}\sum_{i=2^{k-1}}^{2^k}P_i\\&>\frac{1}{2^k}\sum_{i=2^{k-1}}^{2^k}2\cdot 2^{\delta' n}\\&\geq 2^{-\delta' n}\end{aligned}$$

which is a contradiction. Thus $P_{2^{k-1}}\leq 2\cdot 2^{-\delta' n}$.

Now the final code we require has $\mathbf{m}_1,\mathbf{m}_2,\ldots,\mathbf{m}_{2^{k-1}}$ as its messages (and thus has dimension $k'=k-1$). If we have $k\leq\lfloor(n+1)(1-H_{\mathsf{Ber}}(p+\varepsilon))\rfloor$, then we have $k'\leq\lfloor n(1-H_{\mathsf{Ber}}(p+\varepsilon))\rfloor$. Setting $\delta=\delta'+\frac{1}{n}$, we have that for every $\mathbf{m}\in\{0,1\}^{k'}$,

$$\Pr_{\mathbf{e}\sim\mathsf{BSC}_p}[D^*(E^*(\mathbf{m})+\mathbf{e})\neq\mathbf{m}]\leq 2^{-\delta n}.$$

This completes the proof of Shannon's Theorem.

References

[1] Venkatesan Guruswami, Atri Rudra, and Madhu Sudan. *Essential Coding Theory*. https://cse.buffalo.edu/faculty/atri/courses/coding-theory/book/.

[2] Raymond Hill. *A First Course in Coding Theory*. Oxford University Press, 1986.