

# 1 Избыточное кодирование

Избыточное кодирование позволяет восстановить информацию, даже если часть кода была потеряна.

В избыточном кодировании обычно используют код фиксированной длины, так как код переменной длины сделать избыточным крайне сложно.

$$c : \Sigma \rightarrow \mathbb{B}^k$$

**Определение.** Расстояние Хемминга

$x, y$  — строки одинаковой длины.

$$H(x, y) = |\{i \mid x[i] \neq y[i]\}|$$

$$H(001001, 111000) = 3$$

**Определение.**  $c$  обнаруживает  $d$  ошибок, если  $\forall a, b \in \Sigma, a \neq b \quad H(c(a), c(b)) > d$

**Определение.**  $c$  исправляет  $d$  ошибок, если  $\forall a, b \in \Sigma, a \neq b \quad H(c(a), c(b)) > 2d$

**Определение.**  $\rho : X \times X \rightarrow \mathbb{R}^+$  — расстояние, если  $\rho$  удовлетворяет следующим аксиомам:

1.  $\rho(x, y) \Leftrightarrow x = y$
2.  $\rho(x, y) = \rho(y, x)$
3.  $\rho(x, y) + \rho(y, z) \geq \rho(x, z)$

**Лемма 1.**  $H$  — расстояние.

$c$  — исправляет  $d$  ошибок, тогда  $x = c(a), x \mapsto y$ , изменив  $\leq d$  битов.

**Лемма 2.**  $\forall d, \forall \Sigma \quad \exists$  код, исправляющий  $d$  ошибок.

*Доказательство.*  $|\Sigma| = n$

$$k : 2^k \geq n$$

$\triangleleft c_1(a)$  = строка длины  $k$ , соответствующая номеру  $a$  в алфавите  $\Sigma$

$\triangleleft c(a) = c_1(a)c_1(a) \dots c_1(a)$ , всего  $2\alpha + 1$  раз.

Этот код исправляет  $d$  ошибок, почему — хз. Не откажусь от доказательства. □

**Определение.** Шаром радиуса  $r$  с центром  $x$  называется  $B_r(x) = \{y \mid \rho(x, y) \leq r\}$

**Определение.** Булевым шаром называется шар, в котором  $x, y \in \mathbb{B}^n$

**Определение.** Объем булева шара — число векторов, которые в нем содержатся.

$$|B_r(x)| = 1 + n + C_n^2 + \dots + C_n^r = \sum_{i=0}^r C_n^i$$

**Лемма 3.** Если код  $c$  обнаруживает  $d$  ошибок, то шары радиуса  $d$  с центрами в кодовых словах не содержат других кодовых слов.

**Лемма 4.** Если код  $c$  исправляет  $d$  ошибок, то шары радиуса  $d$  с центрами в кодовых словах не пересекаются.

### Теорема 1. Граница Хемминга

Для  $\Sigma$ , содержащего  $s$  символов, построен код  $c : \Sigma \rightarrow \mathbb{B}^n$ , исправляющий  $d$  ошибок.

Тогда

$$2^n \geq s \sum_{i=0}^d C_n^i$$

, в частности для  $d = 1$   $2^n \geq s(n + 1)$

### 1.1 Код Хэмминга (оптимальное кодирование для $d = 1$ )

$$s = 2^k$$

Занумеруем все биты от 1 до  $n$ .

Все биты либо контрольные, либо информационные. Возьмём  $2^i$ -тые биты как контрольные, остальные — информационные. Всего берём столько битов, чтобы набралось  $k$  информационных битов.

Например для  $k = 7$  :  $cc i_1 c i_2 i_3 i_4 c i_5 i_6 i_7$ . Асимптотически контрольных битов  $\log$ .

Контрольный бит с номером  $2^i$  задается так, чтобы  $\bigoplus_{\substack{j=1 \dots n \\ j \& 2^i \neq 0}} x[j] = 0$

Проверка смотрит, что нужный  $\bigoplus = 0$ . Все  $i$ , для которых это не выполняется, суммируются. Бит на полученной позиции был потерян, его нужно поменять.

**Теорема 2.** Код Хэмминга исправляет одну ошибку.

*Доказательство.* Докажем, что  $\forall a, b \in \Sigma, a \neq b \ H(c(a), c(b)) > 2$

Рассмотрим строку с одним различающимся разрядом  $j$ . Тогда различаются хотя бы два контрольных бита, т.к. в двоичном представлении  $j$  есть хотя бы две единицы.

Рассмотрим строку с двумя различающимися разрядами  $j$  и  $k$ . Тогда различается хотя бы один контрольный бит, хз почему.  $\square$

Найдём асимптотику.

Пусть всего есть  $n$  бит, взяли  $\log n$  контрольных и  $n - \log n$  информационных.

$$S \sim 2^{n-\log n} \sim \frac{2^n}{n}$$

**Определение.** Линейный код  $c(a) \oplus c(b) = c(p)$

**Лемма 5.**  $H(x \oplus z, y \oplus z) = H(x, y)$

$H(c(a), c(b)) = H(c(a) \oplus c(a), c(a) \oplus c(b)) = H(0, c(p)) = \omega(c(p))$

**Лемма 6.** Код Хемминга — линейный