

# Дискретная математика 2 семестр ПИ, Лекции

Собрано 6 марта 2022 г. в 11:36

---

## Содержание

<b>1. Кодирование информации</b>	<b>1</b>
1.1. Задача об оптимальном префиксном коде . . . . .	1
1.2. Неравенство Крафта . . . . .	3
1.3. Напоминка . . . . .	4
1.4. Конечная случайная схема . . . . .	4
1.5. Количество информации . . . . .	6

## Раздел #1: Кодирование информации

### 1.1. Задача об оптимальном префиксном коде

Пусть  $\Lambda$  – произвольное конечное множество (алфавит),  $a \in \Lambda$  – символы. Пусть  $\forall a \in \Lambda \exists l(a) \in \mathbb{N}, \exists c(a) = \{0, 1\}^{l(a)}$  – кодовая последовательность  $a$ , где  $l(a)$  – длина.

Очевидно, условие  $\forall a, b \in \Lambda \rightarrow (a \neq b \Rightarrow c(a) \neq c(b))$  не является достаточным для однозначного распознавания символов.

**Def 1.1.1.** Код называется префиксным, если  $\forall a, b \in \Lambda \ c(a) = \omega \Rightarrow \nexists m \in \mathbb{N}_0 : c(b) = \omega\gamma$ , где  $\gamma \in \{0, 1\}^m$

Пусть  $\forall a \in \Lambda$  соответствует вероятность  $p(a)$  появления этого символа в сообщении.  $\sum_{a \in \Lambda} p(a) = 1$  и считаем  $\forall a \in \Lambda \ p(a) > 0$ .

Введем дискретную случайную величину  $l : \forall a \in \Lambda \ Pr\{l = l(a)\} = p(a)$  – длина кодовой последовательности символа в сообщении.

**Def 1.1.2.** Оптимальным называется префиксный код, минимизирующий математическое ожидание  $\mathbb{E}l = \sum_{a \in \Lambda} l(a)p(a)$

Чем чаще встречается символ, тем короче должна быть кодовая последовательность.

Почему вообще ОПК существует? Известно, что  $\mathbb{E}l \geq 1$  (в каждой кодовой последовательности должен быть хотя бы один символ). Всегда можно сделать префиксный код, в котором все символы имеют одинаковые длины кодовых последовательностей и эти последовательности различны ( $\forall a \in \Lambda \ l(a) = \lceil \log_2(|\Lambda|) \rceil$ ), т.е. префиксный код существует и матожидание длины кодовой последовательности ограничено.

**Lm 1.1.3.** Если в некотором коде  $C$  существует  $x \in \Lambda : c(x) = \omega\alpha$ , где  $\alpha \in \{0, 1\}^k$  и при этом  $\nexists y \in \Lambda, y \neq x : c(y) = \omega\gamma$ , где  $\gamma \in \{0, 1\}^k$  (то есть, если  $\omega$  не является началом никакой другой кодовой последовательности, кроме  $c(x)$ ), то код  $C' : c'(x) = \omega, \forall y \in \Lambda, y \neq x \ c' = c(y)$  будет префиксным (по построению и условию леммы) и  $\mathbb{E}l' = \mathbb{E}l - p(x)l(x) + p(x)(l(x) - 1) = \mathbb{E}l - p(x) < \mathbb{E}l$ . Тогда код  $C$  точно не мог быть оптимальным.

**Lm 1.1.4** (Лемма о кратчайшем префиксе). Если в префиксном коде  $C \exists a, b \in \Lambda, a \neq b : p(a) < p(b), l(a) < l(b)$ , то такой код не оптимален.

*Доказательство.* Проверим, что для кода  $C'$ , в котором  $c'(a) = c(b), c'(b) = c(a)$  и  $\forall x \in \Lambda : x \neq a, x \neq b \ c'(x) = c(x)$  верно  $\mathbb{E}l - \mathbb{E}l' > 0$ .

$$\mathbb{E}l - \mathbb{E}l' = p(a)l(a) + p(b)l(b) - p(a)l(b) - p(b)l(a) = (p(a) - p(b))(l(a) - l(b)) > 0$$

■

**Lm 1.1.5** (Лемма о соседстве самых редких символов). Пусть  $a, b \in \Lambda, a \neq b$  – символы с наименьшими вероятностями ( $\forall x \in \Lambda \ p(x) \geq p(b) \geq p(a)$ ). Тогда  $\exists$  ОПК :  $c(a) = \omega 0, c(b) = \omega 1$ , где  $\exists k \in \mathbb{N}_0 : \omega \in \{0, 1\}^k$  и это самые длинные кодовые последовательности.

*Доказательство.* Пусть  $C'$  – ОПК. По лемме о кратчайшем префиксе  $a$  и  $b$  имеют самые длинные кодовые последовательности в  $C'$ :  $\forall x \in \Lambda, x \neq a, x \neq b \ l'(a) \geq l'(b) \geq l'(x)$

Если  $c(a) = \omega\gamma, \omega \in \{0, 1\}^{l'(b)}, \gamma \in \{0, 1\}^{l'(a)-l'(b)}$  и  $\omega$  не является началом никакой кодовой последовательности (т.к. остальные кодовые последовательности не длиннее  $\omega$  и  $\nexists$  символа с кодовой последовательностью  $\omega$  в силу префиксности  $C'$ )  $\Rightarrow$  можно сократить кодовую последовательность  $a$ , создав более оптимальный код (?!).

$\Rightarrow$  из оптимальности  $C'$  следует  $l(a) = l(b)$ . Пусть  $c'(b) = \omega 1$ , тогда, если  $\exists x \in \Lambda : c'(x) = \omega 0$ , то построим ОПК  $C : c(a) = c'(x), c(x) = c'(a), \forall x \in \Lambda, x \neq a, x \neq b \ c(z) = c'(z)$ .

Если  $\nexists x \in \Lambda : c'(x) = \omega 0$ , то построим ОПК  $C : c(a) = \omega 0, \forall x \in \Lambda, x \neq a \ c(z) = c'(z)$ . ■

**Лм 1.1.6** (Лемма об ОПК для расширенного алфавита). Пусть  $a, b \in \Lambda, a \neq b$  – символы с наименьшими вероятностями.  $\Lambda' = \Lambda \setminus \{a, b\} \cup \{\underbrace{ab}_{\notin \Lambda}\}$ , где  $\underbrace{ab}_{\notin \Lambda} \notin \Lambda, p(\underbrace{ab}_{\notin \Lambda}) = p(a) + p(b)$ . Пусть  $C'$  – ОПК для  $\Lambda', c'(\underbrace{ab}_{\notin \Lambda}) = \omega$ . Тогда для  $\Lambda$  код  $C : c(a) = \omega 0, c(b) = \omega 1, \forall x \in \Lambda, x \neq a, x \neq b \ c(x) = c'(x)$  будет ОПК.

*Доказательство.*  $l(a)p(a) + l(b)p(b) = (l'(\underbrace{ab}_{\notin \Lambda}) + 1)(p(a) + p(b)) = l'(\underbrace{ab}_{\notin \Lambda})p(\underbrace{ab}_{\notin \Lambda}) + p(\underbrace{ab}_{\notin \Lambda})$ . Тогда  $\mathbb{E}l = \mathbb{E}l' + p(\underbrace{ab}_{\notin \Lambda})$ .

Пусть  $\bar{C}$  – ОПК для  $\Lambda$  и  $\mathbb{E}\bar{l} < \mathbb{E}l$ . По лемме о соседстве:  $\bar{c}(a) = \gamma 0, \bar{c}(b) = \gamma 1$ . Построим  $\bar{C}'$  для  $\Lambda' : \bar{c}'(\underbrace{ab}_{\notin \Lambda}) = \gamma$  и  $\forall x \in \Lambda, x \neq a, x \neq b \ \bar{c}'(x) = \bar{c}(x)$ .

$\bar{C}'$  – префиксный? По Лемме о кратчайшем префиксе  $\nexists$  символа с кодовой последовательностью длины  $> \bar{l}(a)$ . Никакой символ не мог иметь кодовую последовательность  $\gamma$ , т.к.  $\bar{C}$  префиксный. Единственные две последовательности длины  $\bar{l}(a)$ , начинающиеся на  $\gamma$  – это коды  $a$  и  $b$ . Но их нет в  $\Lambda'$ . При этом  $\mathbb{E}\bar{l} = \mathbb{E}\bar{l}' = p(\underbrace{ab}_{\notin \Lambda})$ . По предположению  $\mathbb{E}l' + p(\underbrace{ab}_{\notin \Lambda}) = \mathbb{E}l > \mathbb{E}\bar{l} = \mathbb{E}\bar{l}' + p(\underbrace{ab}_{\notin \Lambda})$

(?!), оптимальности  $C' \Rightarrow \mathbb{E}\bar{l} \geq \mathbb{E}l$ , но т.к.  $\bar{C}$  – ОПК  $\Rightarrow \mathbb{E}\bar{l} = \mathbb{E}l$  и  $C$  – ОПК. ■

Задача: нужно построить ОПК на алфавите  $\Lambda, |\Lambda| = M$ . По лемме об ОПК для расширенного алфавита задачу построения ОПК можно свести к такой же задаче, но с исходным алфавитом с числом букв на единицу меньше, и с набором вероятностей, получающимся из первоначального сложением двух наименьших вероятностей.

Уменьшаем пока не получится алфавит из двух букв. ОПК для алфавита из 2-х букв –  $\{0, 1\}$ . Строже:  $\Lambda_0 := \Lambda$ .  $\forall k \in 0 \dots (M-3)$  берем  $a_k, b_k \in \Lambda_k : \forall x \in \Lambda_k, x \neq a_k, x \neq b_k \ p(a_k) \leq p(b_k) \leq p(x)$  и построим  $\Lambda_{k+1} = \Lambda_k \setminus \{a_k, b_k\} \cup \{\underbrace{a_k b_k}_{\notin \Lambda_k}\} \dots$

Для  $\Lambda_{M-2} = \{a_{M-2}, b_{M-2}\}$  оптимальным будет код  $C_{M-2} : c_{M-2}(a_{M-2}) = 0, c_{M-2}(b_{M-2}) = 1$ , т.к. для него  $\mathbb{E}l_{M-2} = 1$ .

Теперь для  $k \in 1 \dots (M-2)$  есть ОПК  $C_k$  для  $\Lambda_k$ . По лемме об ОПК для расширенного алфавита строится ОПК  $C_{k-1}$  для  $\Lambda_{k-1}$  такой, что  $c_{k-1}(a_{k-1}) = c_k(a_{k-1}b_{k-1})0, c_{k-1}(b_{k-1}) = c_k(a_{k-1}b_{k-1})1, \forall x \in \Lambda_k, x \neq \underbrace{a_{k-1}b_{k-1}}_{\notin \Lambda_{k-1}} \ c_{k-1}(x) = c_k(x)$ .

Выполняем, пока не получится  $C_0$  – ОПК для  $\Lambda_0 = \Lambda$ .

**Пример 1.1.7.**  $\Lambda_0 = \{a, b, c, d, e, f, g\}, p(a) = 0.13, p(b) = 0.08, p(c) = 0.25, p(d) = 0.18, p(e) = 0.03, p(f) = 0.12, p(g) = 0.21$ .

$a_0 = e, b_0 = b, \Lambda_1 = \{a, \underbrace{e, b}_{\notin \Lambda_1}, c, d, f, g\}, p(a) = 0.13, p(\underbrace{eb}_{\notin \Lambda_1}) = 0.11, p(c) = 0.25, p(d) = 0.18, p(f) =$

$$0.12, p(g) = 0.21.$$

$$a_1 = \underbrace{eb}, b_1 = f, \Lambda_2 = \{a, \underbrace{ebf}, c, d, g\}, p(a) = 0.13, p(\underbrace{ebf}) = 0.23, p(c) = 0.25, p(d) = 0.18, p(g) = 0.21.$$

$$a_2 = a, b_2 = d, \Lambda_3 = \{\underbrace{ad}, \underbrace{ebf}, c, g\}, p(\underbrace{ad}) = 0.31, p(\underbrace{ebf}) = 0.23, p(c) = 0.25, p(g) = 0.21.$$

$$a_3 = g, b_3 = \underbrace{ebf}, \Lambda_4 = \{\underbrace{ad}, \underbrace{gebf}, c\}, p(\underbrace{ad}) = 0.31, p(\underbrace{gebf}) = 0.44, p(c) = 0.25. a_4 = c, b_4 = \underbrace{ad}, \Lambda_5\{\underbrace{cad}, \underbrace{gebf}\}, p(\underbrace{cad}) = 0.56, p(\underbrace{gebf}) = 0.44. \text{ Тогда } c_5(\underbrace{gebf}) = 0, c(\underbrace{cad}) = 1.$$

Теперь раскрываем алфавит обратно:

$$c_4(\underbrace{gebf}) = 0, c_4(c) = 10, c_4(\underbrace{ad}) = 11.$$

$$c_3(g) = 00, c_3(\underbrace{ebf}) = 01, c_3(c) = 10, c_3(\underbrace{ad}) = 11.$$

$$c_1(g) = 00, c_1(\underbrace{eb}) = 010, c_1(f) = 011, c_1(c) = 10, c_1(a) = 110, c_1(d) = 111.$$

$$c_0(g) = 00, c_0(e) = 0100, c_0(b) = 0101, c_0(f) = 011, c_0(c) = 10, c_0(a) = 110, c_0(d) = 111.$$

## 1.2. Неравенство Крафта

Пусть задан набор длин  $l_1, \dots, l_m$ , не все обязательно различны. Может ли такой набор оказаться набором длин некоторого префиксного кода.

**Теорема 1.2.1.** Для того, чтобы набор длин  $l_1, \dots, l_m$  мог быть набором длин кодовых последовательностей некоторого ПК для алфавита из  $m$  символов необходимо и достаточно, чтобы  $\sum_{i=1}^m 2^{-l_i} \leq 1$ .

*Доказательство.* "  $\Rightarrow$  ". Пусть  $\exists$  префиксный код для алфавита с кодовыми последовательностями с длинами  $l_1, \dots, l_m$ . множество кодовых последовательностей – набор всех путей на двоичном дереве от корня к листьям.

Корень – нулевой уровень. Далее последовательно увеличиваем номер по мере удаления от корня.

Каждой вершине  $v$  на уровне  $t$  сопоставим число  $a(v) = 2^{-t}$

Пусть вершина  $v$  на уровне  $t$  – не лист. Т.е. на уровне  $t+1$  есть  $\geq 1$  вершина, получившаяся из  $v$ . Обозначим её  $N(v)$ . Тогда  $a(v) \geq \sum_{u \in N(v)} a(u)$

Просуммируем неравенства для всех не листов:

$$\sum_{v \text{ не лист}} a(v) \geq \sum_{u \text{ не корень}} a(u)$$

$$\Rightarrow 2^0 \geq \sum_{u \text{ листья}} a(u). \text{ Необходимость доказана.}$$

"  $\Leftarrow$  ". Считаем, что выполнено неравенство и пусть  $l_1 \leq l_2 \leq \dots \leq l_m$

$n_j$  – число листьев на уровне  $j$ :  $n_j = |\{i : l_i = j, i \in 1 : m\}|$

$$\sum_{i \in 1:m} 2^{-l_i} \geq 1 \Rightarrow \sum_{j \in 1:l_m} 2^{-j} n_j \leq 1. \text{ Тогда } \forall j \in 1 : l_m : n_j \leq 2^j - (2^{j-1} n_1 + \dots + 2 n_{j-1})$$

Пусть  $m \neq 1$ . Выделим на первом уровне вершин  $n_1 \leq 2$ , на втором уровне останется  $2(2 - n_1)$ . Известно, что  $n_2 \leq 2^2 - 2n_1 \Rightarrow$  осталось не меньше, чем требуется для второго уровня.

$(j-1)$ -уровень: было свободно  $2^{j-1} - (2^{j-2}n_1 + \dots + 2n_{j-2})$  и  $n_{j-1}$  не больше этой величины. Выделим  $n_{j-1}$  узлов, останется  $2^{j-1} - (2^{j-2}n_1 + \dots + 2n_{j-2}) - n_{j-1}$ . Значит на  $j$ -м уровне будет  $2 \cdot (\dots) = 2^j - (2^{j-1}n_1 + \dots + 2n_{j-1})$  ■

### 1.3. Напоминка

Пусть  $S$  – конечное множество.  $|S| = n$ .

Пусть задана функция  $f: S \rightarrow [0, 1]$ ,  $\forall \omega \in S \exists! f(\omega) \in [0, 1]$

$\sum_{\omega \in S} f(\omega) = 1$ . Определим  $\forall A \subseteq S$  величину  $Pr(A) = \sum_{\omega \in A} f(\omega)$

Функция  $f$  в общем-то и не нужна. Достаточно иметь  $Pr$ .

**Def 1.3.1.**  $(S, Pr)$  называется вероятностным пространством.

$S$  – пространство элементарных событий.

$\omega \in S$  – элементарное событие (исход).  $A \subseteq S$  – событие.  $Pr(A)$  – вероятность  $A$ .

$A, B \subseteq S, Pr(A \cap B) = 0$  – несовместные события.

**Свойства вероятности:**

- $Pr(A \cup B) = Pr(A) + Pr(B) - Pr(A \cap B)$
- $Pr(A) + Pr(S \setminus A) = 1$
- $Pr(A \cup B) \leq Pr(A) + Pr(B)$
- $Pr(A) = Pr(A \setminus B) + Pr(A \cap B)$

**Неравенство Йенсена:**

**Def 1.3.2.** Функция  $f$  называется выпуклой на  $X \in R$ , если  $\forall x_1, x_2 \in X$  и  $\forall \alpha \in [0, 1]$  выполняется неравенство  $f(\alpha x_1 + (1 - \alpha)x_2) \leq \alpha f(x_1) + (1 - \alpha)f(x_2)$

Неравенство Йенсена: пусть  $f$  выпуклая на  $X$  функция. Тогда  $f(\sum_{i=1}^n \alpha_i x_i) \leq \sum_{i=1}^n \alpha_i f(x_i)$ , где

$x_i \in X, \alpha_i \geq 0, \sum_{i=1}^n \alpha_i = 1$ .

*Доказательство.* База при  $n = 2$  верна по определению выпуклой функции. Пусть  $f$  – выпуклая на  $X$  функция. Тогда

$$\begin{aligned} f\left(\sum_{i=1}^{n+1} \alpha_i x_i\right) &= f\left((1 - \alpha_{n+1}) \sum_{i=1}^n \frac{\alpha_i}{1 - \alpha_{n+1}} x_i + \alpha_{n+1} x_{n+1}\right) \leq (1 - \alpha_{n+1}) f\left(\sum_{i=1}^n \frac{\alpha_i}{1 - \alpha_{n+1}} x_i\right) + \alpha_{n+1} f(x_{n+1}) \leq \\ &\leq (1 - \alpha_{n+1}) \sum_{i=1}^n \frac{\alpha_i}{1 - \alpha_{n+1}} f(x_i) + \alpha_{n+1} f(x_{n+1}) = \sum_{i=1}^{n+1} \alpha_i f(x_i) \end{aligned}$$

### 1.4. Конечная случайная схема

**Def 1.4.1.** Пусть  $A_1, A_2, \dots, A_n$  – разбиение множества исходов  $S$  вероятностного пространства  $(S, Pr)$ . Конечной случайной схемой называется схема  $\alpha$ , сопоставляющая каждому  $A_i$  вероятность  $Pr(A_i)$

**Def 1.4.2.** Энтропией КСС называется  $H(\alpha) = - \sum_{i=1}^n Pr(A_i) \times \log Pr(A_i)$

**Свойства энтропии:**

- $H(\alpha) \geq 0$
- Энтропия характеризует неопределенность, заключенную в КСС
- Для любой  $\alpha \subset k$  исходами справедливо  $H(\alpha) \leq \log k$

*Доказательство.*  $f(x) := -x \cdot \log x$ . На  $[0, 1]$  функция  $f(x)$  строго вогнутая  $\Rightarrow$  по неравенству Йенсена  $\sum_{i=1}^n \lambda_i \cdot f(x_i) \leq f(\sum_{i=1}^n \lambda_i \cdot x_i)$ , причём равенство  $\Leftrightarrow x_1 = \dots = x_n$ .

$$\begin{aligned} \text{Тогда возьмём } x_i &= Pr(A_i) \text{ и } \lambda_i = \frac{1}{k} \forall i \in 1 \dots k, \text{ получаем } \sum_{i=1}^k \frac{1}{k} (-Pr(A_i) \times \log Pr(A_i)) \leq \\ &\leq - \sum_{i=1}^k \frac{1}{k} Pr(A_i) \times \log \left( \sum_{i=1}^k Pr(A_i) \right) \\ &= - \frac{1}{k} \sum_{i=1}^k Pr(A_i) \times \log Pr(A_i) \leq - \frac{1}{k} \log \frac{1}{k} \\ &= - \sum_{i=1}^k Pr(A_i) \times \log Pr(A_i) \leq \log k \end{aligned}$$

Максимальная энтропия для КСС имеет схема с  $k$  равновероятностными исходами.

$H(\alpha) = 0 \Leftrightarrow \exists!$  достоверный исход в  $\alpha$  ■

Пусть есть КСС  $\alpha$  с исходами  $A_1, \dots, A_k$  и КСС  $\beta$  с исходами  $B_1, \dots, B_l$ . Их пересечением  $\alpha \cap \beta$  называются КСС, исходы которой –  $A_i \cap B_j, \forall i \in 1, \dots, k, j \in 1, \dots, l$

$$\text{Тогда } H(\alpha \cap \beta) = - \sum_{i=1}^k \sum_{j=1}^l Pr(A_i \cap B_j) \times \log Pr(A_i \cap B_j)$$

$$\begin{aligned} \text{Т.к. } Pr(A_i \cap B_j) &= Pr(A_i) \times Pr(B_j|A_i) \Rightarrow H(\alpha \cap \beta) = \\ &= - \sum_{i=1}^k \sum_{j=1}^l Pr(A_i) Pr(B_j|A_i) \times (\log Pr(A_i) + \log Pr(B_j|A_i)) = \\ &= - \sum_{i=1}^k \sum_{j=1}^l Pr(A_i) Pr(B_j|A_i) \times \log Pr(A_i) - \sum_{i=1}^k \sum_{j=1}^l Pr(A_i) Pr(B_j|A_i) \times \log Pr(B_j|A_i) = \\ &= - \sum_{i=1}^k Pr(A_i) \cdot \log Pr(A_i) \cdot \sum_{j=1}^l Pr(B_j|A_i) + \sum_{i=1}^k Pr(A_i) \cdot \left( - \sum_{j=1}^l Pr(B_j|A_i) \cdot \log Pr(B_j|A_i) \right) = \\ &= - \sum_{i=1}^k Pr(A_i) \cdot \log Pr(A_i) + \dots = H(\alpha) + \dots \end{aligned}$$

**Def 1.4.3.** Величину  $H(\beta|A_i) := - \sum_{j=1}^l Pr(A_i) Pr(B_j|A_i) \cdot \log Pr(B_j|A_i)$  называют условной энтропией  $\beta$  при условии  $A_i$

**Def 1.4.4.** Величину  $H_\alpha(\beta) := \sum_{i=1}^k Pr(A_i) \cdot H(\beta|A_i)$  называют средней условной энтропией  $\beta$  при условии  $\alpha$ .

Таким образом,  $H(\alpha \cap \beta) = H(\alpha) + H_\alpha(\beta)$

Докажем, что  $0 \leq H_\alpha(\beta) \leq H(\beta)$

Неотрицательность следует из неотрицательности энтропий.

fix  $j$ ,  $f(x) = -x \cdot \log x$ ,  $\lambda_i = Pr(A_i)$ ,  $x_i = Pr(B_j|A_i) \quad \forall i \in 1...k$

Неравенство Йенсена:  $\sum_{i=1}^k Pr(A_i) \cdot (-Pr(B_j|A_i) \cdot \log Pr(A_i)) \leq$

$$\leq \left( - \sum_{i=1}^k Pr(A_i) \cdot Pr(B_j|A_i) \right) \cdot \log \sum_{i=1}^k Pr(A_i) \cdot Pr(B_j|A_i)$$

$$\text{ПЧ} = \left( - \sum_{i=1}^k Pr(A_i) \cdot Pr(B_j|A_i) \right) \cdot \log \sum_{i=1}^k Pr(A_i) \cdot Pr(B_j|A_i) =$$

$$= - \left( \sum_{i=1}^k Pr(B_j \cap A_i) \right) \cdot \log \sum_{i=1}^k Pr(B_j \cap A_i) = -Pr(B_j) \cdot \log Pr(B_j)$$

Просуммируем по  $j$ :  $\sum_{j=1}^l \sum_{i=1}^k Pr(A_i) \cdot (-Pr(B_j|A_i) \cdot \log Pr(A_i)) \leq \sum_{j=1}^l (-Pr(B_j) \cdot \log Pr(B_j))$

$$\sum_{i=1}^k Pr(A_i) \cdot \sum_{j=1}^l (-Pr(B_j|A_i) \cdot \log Pr(A_i)) \leq - \sum_{j=1}^l Pr(B_j) \cdot \log Pr(B_j)$$

$$\sum_{i=1}^k Pr(A_i) \cdot H(\beta|A_i) \leq H(\beta) \Rightarrow H_\alpha(\beta) \leq H(\beta)$$

$H_\alpha(\beta) = H(\beta) \Leftrightarrow$  все  $Pr(B_j|A_i)$  равны между собой.

Формула полной вероятности:  $\forall j \in 1...l Pr(B_j) = \sum_{i=1}^k Pr(B_j|A_i) \cdot Pr(A_i)$

$$\forall j \in 1...l Pr(B_j) = Pr(B_j|A_1) \cdot \sum_{i=1}^k Pr(A_i) = Pr(B_j|A_1).$$

То есть  $\forall i \in 1...k, j \in 1...l \quad Pr(B_j) = Pr(B_j|A_i)$

**Def 1.4.5.** События  $A$  и  $B$  – взаимно независимы  $\Leftrightarrow Pr(A \cap B) = Pr(A) \cdot Pr(B) \Leftrightarrow Pr(A) \cdot Pr(B|A) = Pr(A) \cdot Pr(B) \Leftrightarrow Pr(B|A) = Pr(B)$

**Def 1.4.6.** КСС  $\alpha$  и  $\beta$  называются независимыми, когда все исходы  $\alpha$  независимы со всеми исходами  $\beta$ . В таком случае  $H_\alpha(\beta)$  максимальна и равна  $H(\beta)$

## 1.5. Количество информации

**Def 1.5.1.** Величина  $I(\alpha, \beta) = H(\beta) - H_\alpha(\beta)$  называется количеством информации.

**Свойства:**

- $I(\alpha, \beta) \geq 0$
- $I(\alpha, \beta) = H(\beta) \Leftrightarrow H_\alpha(\beta) = 0$
- $I(\alpha, \beta) = I(\beta, \alpha)$
- $I(\alpha, \beta) = 0 \Leftrightarrow \alpha$  и  $\beta$  независимы.

**Пример:**

Загадано натуральное число  $x \in 1...N$

$\beta$  – опыт, состоящий в нахождении  $x$ ,  $\beta_m$  – опыт, показывающий, делится ли  $x$  на  $m$ ,  $m \in 1...N$ .

У  $\beta$  есть  $N$  исходов, у  $\beta_m$  — два исхода.

$$H_{\beta_m}(\beta) = Pr(x:m) \cdot H(\beta|x:m) + Pr(x \neq m) \cdot H(\beta|x \neq m)$$

$q := \left\lfloor \frac{N}{m} \right\rfloor$  — количество чисел от 1 до  $N$ , делящихся на  $m$ . Тогда  $Pr(x:m) = \frac{q}{N}$ ,  $Pr(x \neq m) = \frac{N-q}{N}$ .

$$H(\beta|x:m) = - \sum_{i:m, i \in 1 \dots m} \frac{1}{q} \cdot \log \frac{1}{q} = -\frac{q}{q} \cdot \log \frac{1}{q} = \log q.$$

$$\text{Аналогично } H(\beta|x \neq m) = \log(N-q) \Rightarrow H_{\beta_m}(\beta) = \frac{q}{N} \cdot \log q + \frac{N-q}{N} \cdot \log(N-q)$$

$$\begin{aligned} I(\beta_m, \beta) &= \log N - \frac{q}{N} \cdot \log q - \frac{N-q}{N} \cdot \log(N-q) = \\ &= \frac{q}{N} \cdot \log N - \frac{q}{N} \log q - \frac{N-q}{N} \cdot \log N - \frac{N-q}{N} \cdot \log(N-q) = \\ &= -\frac{q}{N} \cdot \log \frac{q}{N} - \frac{N-q}{N} \cdot \log \frac{N-q}{N} \leq \log 2 \end{aligned}$$

Равенство достигается при  $q = N - q = \frac{N}{2}$ .

### Данетки:

Загадано число от 1 до  $N$ .

Опыт  $\beta$  — угадать число.

Опыт  $\alpha$  — задать любой общий (да/нет) вопрос и получить ответ.

$H(\beta) = \log N$  (Числа загаданы с равной вероятностью)

$H(\alpha) \leq \log 2$  (Поскольку есть всего 2 варианта ответа)

$H(\alpha_1 \alpha_2 \dots \alpha_k) \leq \log 2^k = k \log 2$  ( $k$  вопросов, 2 варианта ответа)

Чтобы угадать число потребуется  $k \geq \frac{\log N}{\log 2} = \log_2 N$  вопросов.

Есть ли алгоритм, который умеет угадывать загаданное число за  $O(\log N)$ .

### Избыточное кодирование:

Есть сообщение  $u \in \{0, 1\}^k$ , которое нужно передать.

Можем передавать сообщение  $x(u) \in \{0, 1\}^n$ ,  $n \geq k$ , содержащую некоторую избыточную информацию (канал связи шумит и может допускать ошибки), но не более  $d$  ошибок на сообщение.

$\beta$  заключается в нахождении всех  $d$  ошибок. Сколько у  $\beta$  исходов? Для каждого количества

ошибок  $j$  от 0 до  $d$  есть  $\binom{n}{j}$  вариантов их расположения, то есть всего исходов у  $\beta$  ровно  $\sum_{j=0}^d \binom{n}{j}$

$$\text{Следовательно, } H(\beta) = \log \sum_{j=0}^d \binom{n}{j}$$

$\alpha$  — дополнительное сообщение размера  $n - k$ . Их  $2^{n-k}$  и  $\Rightarrow H(\alpha) = \log 2^{n-k} = (n - k) \log 2$

Чтобы гарантированно найти все ошибки нужно  $H(\alpha) \geq H(\beta)$

$$(n - k) \log 2 \geq \log \sum_{j=0}^d \binom{n}{j} \Rightarrow n - k \geq \log_2 \sum_{j=0}^d \binom{n}{j} \Rightarrow k \leq n - \log_2 \sum_{j=0}^d \binom{n}{j}$$

Таким образом, если канал связи допускает не более  $d$  ошибок, то для передачи сообщения размером  $k$  понадобится не менее  $k + \log_2 \sum_{j=0}^d \binom{n}{j}$ .

Или, поскольку количество ошибок обычно зависит от размера переданного сообщения, если передаётся  $n$  бит и из них не более  $d$  могут быть ошибочными, то в переданном сообщении



можно закодировать сообщение длиной не более  $n - \log_2 \sum_{j=0}^d \binom{n}{j}$ .

### Код Хэмминга:

Предыдущая задача при  $d = 1$ . Известно, что  $2^{n-k} \geq \sum_{j=0}^1 \binom{n}{j} = 1 + n$ .

$l := n - k$  – длина "избыточного сообщения". Тогда  $k \leq 2^l - l - 1$ .

Чем большее сообщение, тем относительно меньше лишней информации. Как передавать дополнительную информацию?

### Пример:

Пусть  $k = 12$  и мы хотим передать сообщение  $u = 101101011100$ . Зарезервируем в сообщении длины 17 места с номерами  $2^i (1, 2, 4, 8, 16)$ , а на остальные позиции запишем сообщение:

$x_0(u) = \_ \_ 1 \_ 011 \_ 0101110 \_ 0$

Подберём на позицию  $2^i$  такую цифру, чтобы произведение  $x(u)$  и  $i$ -й строки матрицы было равно 0. На "неопределённых" позициях в строке с номером  $i$  стоят 0 ( $2^j = 10 \dots 0$ ).

На позиции  $2^i$  в  $i$ -й строке стоит 1.

$? * 1 + \_ * 0 + 1 * 1 + \_ * 0 + 0 * 1 + 1 * 0 + 1 * 1 + \_ * 0 + 0 * 1 + 1 * 0 + 1 * 1 + 1 * 0 + 0 * 1 + \_ * 0 + 0 * 1 =$   
 $= ? + 1 + 1 + 1 = 1 + ? = 0 \Rightarrow ? = 1$ .

Получается  $x_1 = 1 \_ 1 \_ 011 \_ 0101110 \_ 0$

Аналогично делаем для остальных. Итого  $x(u) = 11110110010111000$

Как определять позицию ошибки?  $y = 11110110000111000$

Посчитаем  $A \times y^T = (0, 1, 0, 1, 0)^T$  – двоичная запись позиции с ошибкой.

Старший бит – справа. Почему так?