Линейные коды

.

Денис Осипов

на основе учебного пособия
«Введение в теорию кодирования»
Ф.И. Соловьевой

Основные определения

E^n — множество двоичных векторов длины n. Оно воспринимается в трех ипостасях:

Основные определения

E^n — множество двоичных векторов длины n. Оно воспринимается в трех ипостасях:

1. Метрическое пространство с **метрикой Хэмминга**:

$$d(u, v) = \#\{i : u_i \neq v_i\}$$

Основные определения

E^n — множество двоичных векторов длины n. Оно воспринимается в трех ипостасях:

1. Метрическое пространство с **метрикой Хэмминга**:

$$d(u, v) = \#\{i : u_i \neq v_i\}$$

2. Векторное пространство над $\mathbb{F}_2$

Основные определения

E^n — множество двоичных векторов длины n. Оно воспринимается в трех ипостасях:

1. Метрическое пространство с **метрикой Хэмминга**:

$$d(u, v) = \#\{i : u_i \neq v_i\}$$

2. Векторное пространство над $\mathbb{F}_2$
3. Аффинное пространство над $\mathbb{F}_2$ (множество точек совпадает с множеством векторов)

Группа его изометрических аффинных автоморфизмов $Aut(E^n)$ исчерпывается всеми перестановками из S_n и векторами E^n:

$$Aut(E^n) = \{(v, \pi) \mid v \in E^n,\ \pi \in S_n\},$$

$$(v, \pi)[u] = \pi(u) + v.$$

$w(u) = \#1 \in u$

Доказательство.

Аффинное: $f(u) = Lu + v_0$ $\quad L: E^n \to E^n$

Изометрия: $d(f(u), f(v)) = d(u, v)$

$$LHS = d(Lu + v_0, Lv + v_0) = d(Lu, Lv) = w(Lu - Lv)$$

$$w(L(u - v)) = w(u - v)$$

$$w(Lu) = w(u)$$

$L \in S^n \quad v = v_0$

□

- **Код** — произвольное подмножество $C \subset E^n$.

- **Код** — произвольное подмножество $C \subset E^n$.
- **Кодовые слова** — элементы C.

- **Код** — произвольное подмножество $C \subset E^n$.
- **Кодовые слова** — элементы C.
- **Длина кода** — число n.

- **Код** — произвольное подмножество $C \subset E^n$.
- **Кодовые слова** — элементы C.
- **Длина кода** — число n.
- **Кодовое расстояние** — минимальное расстояние между кодовыми словами кода.

- Коды C и D **эквивалентные**, если существует изометрический аффинный автоморфизм E^n, переводящий один код в другой. То есть, если существует перестановка $\pi \in S_n$ и вектор $v \in E^n$, такой что $\pi(C) + v = D$.

- Коды C и D **эквивалентные**, если существует изометрический аффинный автоморфизм E^n, переводящий один код в другой. То есть, если существует перестановка $\pi \in S_n$ и вектор $v \in E^n$, такой что $\pi(C) + v = D$.
- **Группа автоморфизмов кода** $Aut(C)$: те изоморфизмы, которые оставляют C на месте. То есть, такие (π, v), что $\pi(C) + v = C$. Ясно, что $Aut(C) \leq Aut(E^n)$.

- Коды C и D **эквивалентные**, если существует изометрический аффинный автоморфизм E^n, переводящий один код в другой. То есть, если существует перестановка $\pi \in S_n$ и вектор $v \in E^n$, такой что $\pi(C) + v = D$.
- **Группа автоморфизмов кода** $Aut(C)$: те изоморфизмы, которые оставляют C на месте. То есть, такие (π, v), что $\pi(C) + v = C$. Ясно, что $Aut(C) \leq Aut(E^n)$.
- **Группа симметрий кода** $Sym(C)$: перестановки, оставляющие код на месте. То есть, $\pi(C) = C$. Ясно, что $Sym(C) \leq Aut(C)$.

- **Линейный код** — код, являющийся линейным подпространством в E^n.

- **Линейный код** — код, являющийся линейным подпространством в E^n.
- $[n, k, d]$-**код** — линейный код в E^n размерности k с кодовым расстоянием d.

- **Линейный код** — код, являющийся линейным подпространством в E^n.
- $[n, k, d]$-**код** — линейный код в E^n размерности k с кодовым расстоянием d.
- $(n, |C|, d)$-**код** — нелинейный код в E^n мощности $|C|$ с кодовым расстоянием d.

Двоичный симметричный канал связи

Пусть при посылке 0 принимается как 0, а 1 как 1, но иногда 0 может быть принят как 1 или 1 принята как 0.

Двоичный симметричный канал связи

Пусть при посылке 0 принимается как 0, а 1 как 1, но иногда 0 может быть принят как 1 или 1 принята как 0.

Для каждого символа имеется вероятность p того, что в канале связи произойдет ошибка

Двоичный симметричный канал связи

Пусть при посылке 0 принимается как 0, а 1 как 1, но иногда 0 может быть принят как 1 или 1 принята как 0.

Для каждого символа имеется вероятность p того, что в канале связи произойдет ошибка , т. е. для переходных (условных) вероятностей $P(\beta \mid \alpha)$, где $\sum_{\alpha \in A} P(\beta \mid \alpha) = 1$, имеем

$$P(0 \mid 0) = P(1 \mid 1) = 1 - p \text{ и } P(1 \mid 0) = P(0 \mid 1) = p.$$

Двоичный симметричный канал связи

Пусть при посылке 0 принимается как 0, а 1 как 1, но иногда 0 может быть принят как 1 или 1 принята как 0.

Для каждого символа имеется вероятность p того, что в канале связи произойдет ошибка , т. е. для переходных (условных) вероятностей $P(\beta \mid \alpha)$, где $\sum_{\alpha \in A} P(\beta \mid \alpha) = 1$, имеем

$$P(0 \mid 0) = P(1 \mid 1) = 1 - p \text{ и } P(1 \mid 0) = P(0 \mid 1) = p.$$

Таким образом, отправленный вектор x и принятый вектор y могут отличаться: $y = x + e$, где e — вектор ошибок. Тогда p — это вероятность того, что $e_i = 1$.

Линейные коды

Способы задания линейного кода:

Линейные коды

Способы задания линейного кода:

1. **Кодовая матрица** $2^k \times n$: просто записываем все кодовые слова в строчки

Линейные коды

Способы задания линейного кода:

1. **Кодовая матрица** $2^k \times n$: просто записываем все кодовые слова в строчки
2. **Порождающая матрица** $k \times n$: записываем базис кода в строчки

Линейные коды

Способы задания линейного кода:

1. **Кодовая матрица** $2^k \times n$: просто записываем все кодовые слова в строчки
2. **Порождающая матрица** $k \times n$: записываем базис кода в строчки
3. **Проверочная матрица** $(n-k) \times n$: код представляется ядром некоторой матрицы.

$$x \in C \Leftrightarrow Hx = 0$$

Каноническая проверочная матрица

Пусть C — $[n, k, d]$-линейный код. Его проверочная матрица H называется **канонической**, если она имеет вид

$$H = [\underset{(n-k)\times k}{A} | \underset{(n-k)\times(n-k)}{E}].$$

Каноническая проверочная матрица

Пусть C — $[n, k, d]$-линейный код. Его проверочная матрица H называется **канонической**, если она имеет вид

$$H = [\underset{(n-k)\times k}{A} | \underset{(n-k)\times(n-k)}{E}].$$

Теорема

Проверочная матрица каноническая $\Longleftrightarrow$ *порождающая матрица имеет вид*

$$G = [\underset{k\times k}{E} | \underset{k\times(n-k)}{-A^T}]$$

Теорема

Проверочная матрица $H = [A_{(n-k)\times k}|E_{n-k}]$
$\iff$ *порождающая матрица* $G = [E_k| - A^T_{k\times(n-k)}]$.

Доказательство.

По опред H $\forall x \in C$ $Hx = 0$

$$\forall i = 1 \dots n-k \quad \sum_{j=1}^{n} h_{ij} x_j = 0 \iff \sum_{j=1}^{k} a_{ij} x_j + x_{k+i} = 0$$

$$x_{k+i} = -\sum_{j=1}^{k} a_{ij} x_j$$

$$x = \begin{pmatrix} x_1 \\ \vdots \\ x_{k+1} \\ \vdots \\ x_n \end{pmatrix}^T = \begin{pmatrix} 1 & & & & & & -a_{11} & \cdots & -a_{1k} \\ & 1 & & & & & -a_{12} & & \\ & & 1 & & & & \vdots & & \\ & & & 1 & & & & & \\ & & & & 1 & & & & \\ & & & & & 1 & & & \end{pmatrix}^T \begin{pmatrix} x_1 \\ \vdots \\ x_k \end{pmatrix}$$

$$= [E_k | -A^T] \begin{pmatrix} x_1 \\ \vdots \\ x_k \end{pmatrix}$$

□

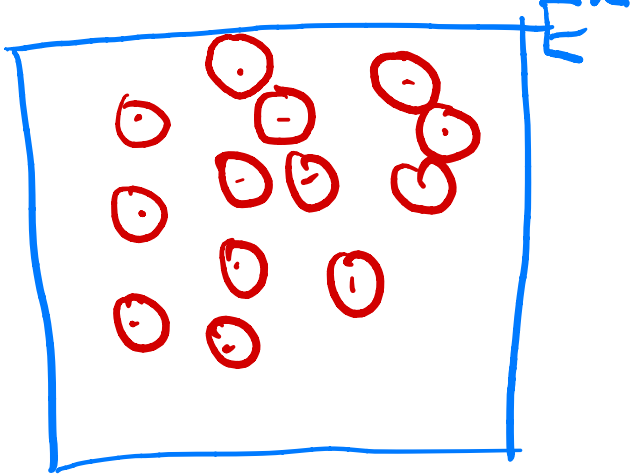
Границы объемов кодов

Теорема (Граница Хэмминга)

Для любого двоичного кода C длины n (не обязательно линейного) с кодовым расстоянием d выполняется неравенство

$$|C| \leq \frac{2^n}{\sum_{i=0}^{[(d-1)/2]} C_n^i}.$$

Доказательство.



$S_r(w) = C_n^0 + C_n^1 + \dots + C_n^r$

$|C| \cdot S_{\left[\frac{d-1}{2}\right]} \leq 2^n$

□

Теорема (Граница Синглтона)

Для любого двоичного $[n, k, d]$-кода выполняется $n - k \geq d - 1$.

Теорема (Граница Синглтона)

Для любого двоичного $[n, k, d]$-кода выполняется $n - k \geq d - 1$.

Эта теорема непосредственно вытекает из следующей:

Теорема (О столбцах проверочной матрицы)

Если H — проверочная матрица кода длины n, то:

кодовое расстояние d $\iff$ *любые $d-1$ столбцов матрицы H линейно независимые,* ***и*** *найдутся d линейно зависимых столбцов.*

Теорема (Граница Синглтона)

Для любого двоичного $[n, k, d]$-кода выполняется $n - k \geq d - 1$.

Эта теорема непосредственно вытекает из следующей:

Теорема (О столбцах проверочной матрицы)

Если H — проверочная матрица кода длины n, то:

кодовое расстояние d $\iff$ *любые $d-1$ столбцов матрицы H линейно независимые, **и** найдутся d линейно зависимых столбцов.*

Доказательство границы Синглтона.

Ранг матрицы H равен $d - 1$, но с другой стороны не превосходит $n - k$.

Доказательство теоремы о столбцах

Лемма $\min\limits_{\substack{x\neq y\\ x,y\in C}} d(x,y) = \min\limits_{\substack{x\neq y\\ x,y\in C}} w(x-y) = \min\limits_{\substack{t\neq 0\\ t\in C}} w(t)$ □

Д-во теоремы код.расст. $d \Leftrightarrow$ $\exists$ код.слово веса d — и
$\nexists$ код.слово веса $\leqslant d$

$Hu = 0 \Leftrightarrow (h_1 \dots h_n)u = 0$ ← дедушка ← наши d зависс. столбцов

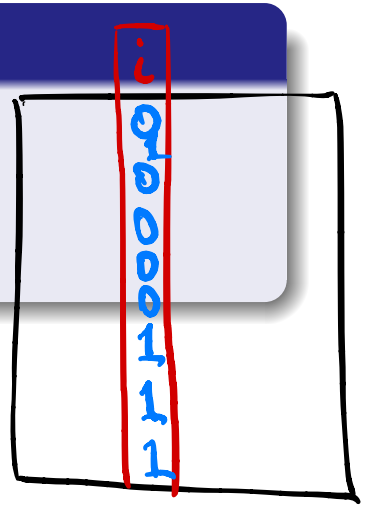
Аналогично про веса $\leqslant d$.

□

Граница Плоткина

Теорема

При $n < 2d$ для любого двоичного (n, M, d)-кода C справедливо неравенство $M \le 2\left[\frac{d}{2d-n}\right]$.



Док-во.

$$S = \sum_{\substack{(x,y)\in C^2 \\ x\neq y}} d(x,y)$$

$\operatorname{argmax}\, x(M-x) = \begin{cases} M \text{ чётно} \Rightarrow \frac{M}{2} \\ M \text{ нечётно} \Rightarrow \frac{M+1}{2} \end{cases}$

① $S \ge d \cdot M(M-1)$

② $S = \sum_{i=1}^{n} 2 n_i (M - n_i)$

n_i = #0 в i-ой коорд. код. мн-ва

M чётное $\Rightarrow S \le n \cdot 2 \cdot \frac{M}{2} \cdot \frac{M}{2} = \frac{nM^2}{2}$

M нечётное $\Rightarrow S \le n \cdot 2 \cdot \frac{M+1}{2} \cdot \frac{M-1}{2} = \frac{n(M^2-1)}{2}$

$\frac{nM^2}{2} \ge d \cdot M(M-1) \Rightarrow \frac{M}{2} \le \left[\frac{d}{2d-n}\right]$

$M \le \left[\frac{n}{2d-n}\right] = \left[\frac{2d}{2d-n}\right] - 1$

$\le 2\left[\frac{d}{2d-n}\right]$

$[2x] - 1 \le 2[x]$

□

Граница Варшамова-Гилберта

Теорема

Если выполняется неравенство

$$1 + C_{n-1}^{1} + \ldots + C_{n-1}^{d-2} < 2^{r},$$

то существует линейный код длины n с кодовым расстоянием $\geq d$ коразмерности $\leq r$. $n-k \leq r$

То есть, $[n, k, d']$-код, где $k \geq n - r$ и $d' \geq d$.

Д-во. Индукция по кол-ву постр. столбцов i.

База $i = d-1$ $\Rightarrow$ выбираем любые незав. ст-ц.

Переход Из i столбцов можно построить $\leq C_i^1 + C_i^2 + \ldots + C_i^{d-2}$ разных лин. комб.

Если $C_i^1 + C_i^2 + \ldots + C_i^{d-2} < 2^r - 1$, то можем взять r_i не равный ни одной из этих комбинаций $\Rightarrow$ он независим с любыми $d-2$ векторами

Зачем думать про размерности и коразмерности?

Рассматривая линейный код длины n и размерности k, обычно подразумевают следующее:

Зачем думать про размерности и коразмерности?

Рассматривая линейный код длины n и размерности k, обычно подразумевают следующее:

- Первые k символов кодового слова несут смысл сообщения, т.н. *информационные биты*

Зачем думать про размерности и коразмерности?

Рассматривая линейный код длины n и размерности k, обычно подразумевают следующее:

- Первые k символов кодового слова несут смысл сообщения, т.н. *информационные биты*
- Оставшиеся $n - k$ символов называются *проверочные биты* и служат для защиты от помех.

Зачем думать про размерности и коразмерности?

Рассматривая линейный код длины n и размерности k, обычно подразумевают следующее:

- Первые k символов кодового слова несут смысл сообщения, т.н. *информационные биты*
- Оставшиеся $n - k$ символов называются *проверочные биты* и служат для защиты от помех.

Таким образом, размерность k отвечает за информативность кодового слова, а коразмерность $r = n - k$ — за степень защищенности кода от помех.

Код Хэмминга

Зафиксируем число m и построим проверочную матрицу, состоящую из всех $2^m - 1$ ненулевых векторов длины m.

Код Хэмминга

Зафиксируем число m и построим проверочную матрицу, состоящую из всех $2^m - 1$ ненулевых векторов длины m. Любые два столбца этой матрицы линейно независимы. Воспользуемся теоремой о столбцах

Код Хэмминга

Зафиксируем число m и построим проверочную матрицу, состоящую из всех $2^m - 1$ ненулевых векторов длины m.
Любые два столбца этой матрицы линейно независимы.
Воспользуемся теоремой о столбцах

$\Longrightarrow$ кодовое расстояние $d = 3$

Код Хэмминга

Зафиксируем число m и построим проверочную матрицу, состоящую из всех $2^m - 1$ ненулевых векторов длины m.
Любые два столбца этой матрицы линейно независимы.
Воспользуемся теоремой о столбцах

$\Longrightarrow$ кодовое расстояние $d = 3$

$\Longrightarrow$ код исправляет одну ошибку.

Код Хэмминга

Зафиксируем число m и построим проверочную матрицу, состоящую из всех $2^m - 1$ ненулевых векторов длины m.
Любые два столбца этой матрицы линейно независимы.
Воспользуемся теоремой о столбцах

$\Longrightarrow$ кодовое расстояние $d = 3$

$\Longrightarrow$ код исправляет одну ошибку.

Код с такой проверочной матрицей называется *кодом Хэмминга* $\mathcal{H}^n$.

Код Хэмминга

Зафиксируем число m и построим проверочную матрицу, состоящую из всех $2^m - 1$ ненулевых векторов длины m.
Любые два столбца этой матрицы линейно независимы.
Воспользуемся теоремой о столбцах

$\Longrightarrow$ кодовое расстояние $d = 3$

$\Longrightarrow$ код исправляет одну ошибку.

Код с такой проверочной матрицей называется *кодом Хэмминга* $\mathcal{H}^n$.
Его параметры:

- Длина кода $n = 2^m - 1$
- Размерность $k = n - m$
- Кодовое расстояние $d = 3$

Код Хэмминга *совершенный*: он достигает границы Хэмминга, т.е.

$$|\mathcal{H}^n| = \frac{2^n}{\sum_{i=0}^{[(d-1)/2]} C_n^i}$$

Доказательство.

$d = 3$ $\quad RHS = \frac{2^n}{C_n^0 + C_n^1} = \frac{2^n}{n+1}$

$$LHS = 2^{n-m} = \frac{2^n}{n+1}$$

□

Пример кода Хэмминга. Декодирование.

Рассмотрим код Хэмминга с параметром $m = 3$ (т.е. длины 7). Расположем столбцы проверочной матрицы в порядке возрастания десятичных значений столбцов.

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

Пример кода Хэмминга. Декодирование.

Рассмотрим код Хэмминга с параметром $m = 3$ (т.е. длины 7). Расположем столбцы проверочной матрицы в порядке возрастания десятичных значений столбцов.

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

Пусть Алиса отправила вектор x, а Боб получил вектор y. Напомним, что кодовое расстояние равно 3.

Пример кода Хэмминга. Декодирование.

Рассмотрим код Хэмминга с параметром $m = 3$ (т.е. длины 7). Расположем столбцы проверочной матрицы в порядке возрастания десятичных значений столбцов.

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

Пусть Алиса отправила вектор x, а Боб получил вектор y.
Напомним, что кодовое расстояние равно 3.
Если Боб видит, что $y \in \mathcal{H}_n$, то $y = x$.

Пример кода Хэмминга. Декодирование.

Рассмотрим код Хэмминга с параметром $m = 3$ (т.е. длины 7). Расположем столбцы проверочной матрицы в порядке возрастания десятичных значений столбцов.

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

Пусть Алиса отправила вектор x, а Боб получил вектор y. Напомним, что кодовое расстояние равно 3.
Если Боб видит, что $y \in \mathcal{H}_n$, то $y = x$. Иначе вектор ошибки $e = y - x$ содержит ровно одну единицу — пусть на i-той позиции.

Пример кода Хэмминга. Декодирование.

Рассмотрим код Хэмминга с параметром $m = 3$ (т.е. длины 7). Расположем столбцы проверочной матрицы в порядке возрастания десятичных значений столбцов.

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

Пусть Алиса отправила вектор x, а Боб получил вектор y. Напомним, что кодовое расстояние равно 3.
Если Боб видит, что $y \in \mathcal{H}_n$, то $y = x$. Иначе вектор ошибки $e = y - x$ содержит ровно одну единицу — пусть на i-той позиции. Заметим, что

$$S_y = Hy = H(x + e) = He$$

равен i-тому столбцу матрицы H.

Пример кода Хэмминга. Декодирование.

Рассмотрим код Хэмминга с параметром $m = 3$ (т.е. длины 7). Расположем столбцы проверочной матрицы в порядке возрастания десятичных значений столбцов.

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

Пусть Алиса отправила вектор x, а Боб получил вектор y. Напомним, что кодовое расстояние равно 3.
Если Боб видит, что $y \in \mathcal{H}_n$, то $y = x$. Иначе вектор ошибки $e = y - x$ содержит ровно одну единицу — пусть на i-той позиции. Заметим, что

$$Hy = H(x + e) = He$$

равен i-тому столбцу матрицы H. А этот столбец — двоичная запись числа i. Мы получили номер искаженной координаты.

Построение новых кодов

1. **Комбинирование.** Пусть G_1, G_2 — порождающие матрицы для $[n_1, k, d_1]$ и $[n_2, k, d_2]$-кодов соответственно. Тогда коды с порождающими матрицами

$$\begin{pmatrix} G_1 & 0 \\ 0 & G_2 \end{pmatrix} \text{ и } (G_1 \mid G_2)$$

представляют собой $[n_1 + n_2, 2k, \min\{d_1, d_2\}]$ и $[n_1 + n_2, k, d]$-коды соответственно, причем $d \geq d_1 + d_2$.

Построение новых кодов

1. **Комбинирование.** Пусть G_1, G_2 — порождающие матрицы для $[n_1, k, d_1]$ и $[n_2, k, d_2]$-кодов соответственно. Тогда коды с порождающими матрицами

$$\begin{pmatrix} G_1 & 0 \\ 0 & G_2 \end{pmatrix} \text{ и } (G_1 \mid G_2)$$

представляют собой $[n_1 + n_2, 2k, \min\{d_1, d_2\}]$ и $[n_1 + n_2, k, d]$-коды соответственно, причем $d \geq d_1 + d_2$.

2. **Добавление общей проверки на четность.** Удлиним код на один символ: добавим к каждому кодовому слову четность его веса. Тогда и кодовое расстояние кода увеличится на единицу.

Построение новых кодов

1. **Комбинирование.** Пусть G_1, G_2 — порождающие матрицы для $[n_1, k, d_1]$ и $[n_2, k, d_2]$-кодов соответственно. Тогда коды с порождающими матрицами

$$\begin{pmatrix} G_1 & 0 \\ 0 & G_2 \end{pmatrix} \text{ и } (G_1 \mid G_2)$$

представляют собой $[n_1 + n_2, 2k, \min\{d_1, d_2\}]$ и $[n_1 + n_2, k, d]$-коды соответственно, причем $d \geq d_1 + d_2$.

2. **Добавление общей проверки на четность.** Удлиним код на один символ: добавим к каждому кодовому слову четность его веса. Тогда и кодовое расстояние кода увеличится на единицу.
3. **Выкалывание координаты.**

Построение новых кодов

1. **Комбинирование.** Пусть G_1, G_2 — порождающие матрицы для $[n_1, k, d_1]$ и $[n_2, k, d_2]$-кодов соответственно. Тогда коды с порождающими матрицами

$$\begin{pmatrix} G_1 & 0 \\ 0 & G_2 \end{pmatrix} \text{ и } (G_1 \mid G_2)$$

представляют собой $[n_1 + n_2, 2k, \min\{d_1, d_2\}]$ и $[n_1 + n_2, k, d]$-коды соответственно, причем $d \geq d_1 + d_2$.
2. **Добавление общей проверки на четность.** Удлиним код на один символ: добавим к каждому кодовому слову четность его веса. Тогда и кодовое расстояние кода увеличится на единицу.
3. **Выкалывание координаты.**
4. **Пополнение кода.** Пусть C — $[n, k, d]$-код. Если найдется вектор a такой, что $d(C, a) \geq d$, то $C \cup (C + a)$ — $[n, k + 1, d]$-код.

Конструкция Плоткина

Теорема

Пусть C, D — нелинейные коды с кодовыми расстояниями d_1 и d_2 соответственно. Тогда множество

$$C^{2n} = \{(x, x+y) \mid x \in C, y \in D\}$$

является нелинейным кодом с кодовым расстоянием $\min\{2d_1, d_2\}$.

Эта конструкция позволяет получить из маленьких кодов с оптимальными [illegible] новые коды с такими же хорошими параметрами.

P-bo $d((x,x+y),(x',x'+y')) = d(x,x') + d(x+y,x'+y')$

если $y = y'$ $\quad = 2d(x,x') \geq 2d_1$

если $y \neq y'$ $\quad \geq d(x,x') + d(y,y') - d(x,x') = d(y,y') \geq d_2$ □

Теорема Глаголева

Теорема

Для любого линейного $[n, k, d]$-кода C существует линейный код C' с теми же параметрами такой, что его базовое множество состоит из кодовых слов минимального веса d.

Д-во Пусть $\{a_1, \dots, a_t\} \subset C$ – макс. незав. сис-во в C, у которого все веса d. Предп-м что $t < k$.

Пусть $b_1 \notin \mathrm{Lin}\{a_1, \dots, a_t\}$ — с мин. весом. $w(b_1) = \tilde{d} > d$.

b_1' – b_1, у которого заменили $\tilde{d} - d$ единиц на нули.

Пусть $\{a_1, \dots, a_t, b_1, b_2, \dots, b_{t-k}\}$ – базис C.

$C' = \mathrm{Lin}\{a_1, \dots, a_t, b_1', b_2, \dots, b_{t-k}\}$.

1) C' — код с мин. весом d: $\mathrm{Lin}\{a_1, \dots, a_t\}$ не менялась

В дополнении $\mathrm{Lin}\{a_1, \dots, a_t\}$ слова потеряли не более $\tilde{d} - d$ единиц

2) C' — разм-ти k.

От противного: $b_1' \in \mathrm{Lin}\{a_1, \dots, a_t, b_2, \dots, b_{t+k}\}$.

$\Rightarrow b_1' \in C$. Кроме того, $w(b_1') < d$, $w(b_1 - b_1') < d$.

$\Rightarrow$ b_1' и $b_1 - b_1' \in \mathrm{Lin}\{a_1, \dots, a_t\} \Rightarrow b_1 \in \mathrm{Lin}\{a_1, \dots, a_t\}$

противоречие.

Повторяем процедуру до k векторов basis

□

Theorem: Any linear code $\mathscr{C} \subset \mathbb{F}_q^n$ of dimension k and minimum weight d can be transformed into a code $\mathscr{D} \subset \mathbb{F}_q^n$ with the same parameters such that $\mathscr{D}$ possesses a basis of weight d vectors.

Proof: In the sequel, the linear subspace of $\mathbb{F}_q^n$ spanned by a set of vectors $\boldsymbol{x}, \boldsymbol{y}, \cdots, \boldsymbol{z} \in \mathbb{F}_q^n$ will be denoted by $\langle \boldsymbol{x}, \boldsymbol{y}, \cdots, \boldsymbol{z} \rangle$. Let $\{\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_t\} \subset \mathscr{C}$ be a maximal set of independent codewords of weight d. Suppose that $t < k$. All codewords in the complement of the span $\langle \boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_t \rangle$ of the $\boldsymbol{a}_i$ have weight $> d$. Pick a codeword $\boldsymbol{b}_1 \notin \langle \boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_t \rangle$ of lowest weight, say $\tilde{d}$, and extend $\{\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_t, \boldsymbol{b}_1\}$ to a basis $\{\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_t, \boldsymbol{b}_1, \cdots, \boldsymbol{b}_{k-t}\}$ of the code $\mathscr{C}$. Now change $\boldsymbol{b}_1$ into a vector $\boldsymbol{b}_1'$ of weight d by changing $\tilde{d} - d$ of the nonzero coordinates into zero ones. Then, linear subspace

$$\mathscr{C}' := \langle \boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_t, \boldsymbol{b}_1', \cdots, \boldsymbol{b}_{k-t} \rangle \subset \mathbb{F}_q^n$$

is a code of minimum weight d, because $\langle \boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_t \rangle$ is unaltered and the words of $\mathscr{C} \setminus \langle \boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_t \rangle$ have changed in at most $\tilde{d} - d$ coordinates. We claim that the dimension of $\mathscr{C}'$ is equal to k. For if $\dim \mathscr{C}' < k$, then $\boldsymbol{b}_1'$ would be a linear combination of the vectors $\boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_t, \boldsymbol{b}_2, \cdots, \boldsymbol{b}_{k-t}$, and, thus, would be an element of the original code $\mathscr{C}$. Since the weight of both $\boldsymbol{b}_1'$ and $\boldsymbol{b}_1 - \boldsymbol{b}_1'$ is smaller than $\tilde{d}$, these vectors would in fact be contained in the linear subspace $\langle \boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_t \rangle$ which contradicts the fact that $\boldsymbol{b}_1 \in \mathscr{C} \setminus \langle \boldsymbol{a}_1, \boldsymbol{a}_2, \cdots, \boldsymbol{a}_t \rangle$. So $\mathscr{C}'$ has the same parameters as $\mathscr{C}$ has, but the maximum number of independent weight d codewords in $\mathscr{C}'$ exceeds that of $\mathscr{C}$. The induction process is obvious. □

Декодирование двоичных кодов

Задача декодера в модели двоичной симметричной связи — по принятому вектору y решить, какое кодовое слово x было отправлено.

Декодирование двоичных кодов

Задача декодера в модели двоичной симметричной связи — по принятому вектору y решить, какое кодовое слово x было отправлено.

Введем вектор ошибок $e = y - x$. Вероятность конкретного вектора ошибок зависит от его веса и равна

$$\mathbb{P}(e = v) = p^{w(v)} \cdot (1-p)^{1-w(v)}.$$

$p < \frac{1}{2}$

Декодирование двоичных кодов

Задача декодера в модели двоичной симметричной связи — по принятому вектору y решить, какое кодовое слово x было отправлено.
Введем вектор ошибок $e = y - x$. Вероятность конкретного вектора ошибок зависит от его веса и равна

$$\mathbb{P}(e = v) = p^{w(v)} \cdot (1-p)^{1-w(v)}.$$

Так как $p < 1/2$, векторы ошибок меньшего веса всегда будут более вероятны, чем векторы большего веса.

Декодирование двоичных кодов

Задача декодера в модели двоичной симметричной связи — по принятому вектору y решить, какое кодовое слово x было отправлено.
Введем вектор ошибок $e = y - x$. Вероятность конкретного вектора ошибок зависит от его веса и равна

$$\mathbb{P}(e = v) = p^{w(v)} \cdot (1-p)^{1-w(v)}.$$

Так как $p < 1/2$, векторы ошибок меньшего веса всегда будут более вероятны, чем векторы большего веса.
Декодирование по максимуму правдоподобия заключается в выборе вектора e наименьшего веса такого, что $y - e$ является кодовым словом.

Декодирование двоичных кодов

Задача декодера в модели двоичной симметричной связи — по принятому вектору y решить, какое кодовое слово x было отправлено.

Введем вектор ошибок $e = y - x$. Вероятность конкретного вектора ошибок зависит от его веса и равна

$$\mathbb{P}(e = v) = p^{w(v)} \cdot (1 - p)^{1 - w(v)}.$$

Так как $p < 1/2$, векторы ошибок меньшего веса всегда будут более вероятны, чем векторы большего веса.

Декодирование по максимуму правдоподобия заключается в выборе вектора e наименьшего веса такого, что $y - e$ является кодовым словом.

Реализация этой стратегии «в лоб» будет слишком медленной из-за того, что нужно проверять все векторы $e \in E^n$.

Декодирование линейных кодов

Пусть C — линейный двоичный $[n, k]$-код. Пространство E^n разбивается на смежные классы по коду C:

$$E = C \sqcup (a^1 + C) \sqcup \ldots \sqcup (a^{2^{n-k}} + C).$$

Вектор y, принятый декодером, попадает в некоторый класс смежности $y \in a^i + C$.

Декодирование линейных кодов

Пусть C — линейный двоичный $[n, k]$-код. Пространство E^n разбивается на смежные классы по коду C:

$$E = C \sqcup (a^1 + C) \sqcup \ldots \sqcup (a^{2^{n-k}} + C).$$

Вектор y, принятый декодером, попадает в некоторый класс смежности $y \in a^i + C$. Так как отправленный вектор $x \in C$ был кодовым словом, то вектор ошибок $e = y - x \in a^i + C$ будет в том же смежном классе.

Декодирование линейных кодов

Пусть C — линейный двоичный $[n, k]$-код. Пространство E^n разбивается на смежные классы по коду C:

$$E = C \sqcup (a^1 + C) \sqcup \ldots \sqcup (a^{2^{n-k}} + C).$$

Вектор y, принятый декодером, попадает в некоторый класс смежности $y \in a^i + C$. Так как отправленный вектор $x \in C$ был кодовым словом, то вектор ошибок $e = y - x \in a^i + C$ будет в том же смежном классе. Поэтому в качестве e можно просто взять элемент $a^i + C$ с наименьшим весом.

Пример

Рассмотрим линейный $[4, 2]$-код с порождающей матрицей

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

Пример

Рассмотрим линейный $[4, 2]$-код с порождающей матрицей

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

Выпишем весь код и его смежные классы в таблицу (*т.н. стандартное расположение кода*)

C	0000	1011	0101	1110

Пример

Рассмотрим линейный $[4, 2]$-код с порождающей матрицей

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

Выпишем весь код и его смежные классы в таблицу (*т.н. стандартное расположение кода*)

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

Пример

Рассмотрим линейный $[4, 2]$-код с порождающей матрицей

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

Выпишем весь код и его смежные классы в таблицу (*т.н. стандартное расположение кода*)

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

Как действует декодер?

Пример

Рассмотрим линейный $[4, 2]$-код с порождающей матрицей

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

Выпишем весь код и его смежные классы в таблицу (*т.н. стандартное расположение кода*)

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

Как действует декодер?

$$y = 1101$$

Пример

Рассмотрим линейный $[4, 2]$-код с порождающей матрицей

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

Выпишем весь код и его смежные классы в таблицу (*т.н. стандартное расположение кода*)

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

Как действует декодер?

$$y = 1101 \implies C + 1000$$

Пример

Рассмотрим линейный $[4, 2]$-код с порождающей матрицей

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

Выпишем весь код и его смежные классы в таблицу (*т.н. стандартное расположение кода*)

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

Как действует декодер?

$$y = 1101 \implies C + 1000 \implies e = 1000$$

Пример

Рассмотрим линейный $[4, 2]$-код с порождающей матрицей

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

Выпишем весь код и его смежные классы в таблицу (*т.н. стандартное расположение кода*)

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

Как действует декодер?

$$y = 1101 \implies C + 1000 \implies e = 1000 \implies x = 0101$$

Пример

Рассмотрим линейный $[4, 2]$-код с порождающей матрицей

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

Выпишем весь код и его смежные классы в таблицу (*т.н. стандартное расположение кода*)

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

Как действует декодер?

$$y = 1101 \implies C + 1000 \implies e = 1000 \implies x = 0101$$

Всё еще неэффективно — нужно строить большую таблицу, и потом по ней искать.

Синдромы

Пусть H — проверочная матрица, y — вектор.

Синдром вектора y — это вектор

$$S_y = Hy.$$

Синдромы

Пусть H — проверочная матрица, y — вектор.
Синдром вектора y — это вектор

$$S_y = Hy.$$

Свойства синдрома:

1. $S_x = 0 \iff x \in C$

Синдромы

Пусть H — проверочная матрица, y — вектор.
Синдром вектора y — это вектор

$$S_y = Hy.$$

Свойства синдрома:

1. $S_x = 0 \iff x \in C$
2. S_y — сумма тех столбцов матрицы H, где произошли ошибки. $H = (h_1 \dots h_n)$ $S_y = Hy = H(x+e) = He$ □

Синдромы

Пусть H — проверочная матрица, y — вектор.
Синдром вектора y — это вектор

$$S_y = Hy.$$

Свойства синдрома:

1. $S_x = 0 \iff x \in C$
2. S_y — сумма тех столбцов матрицы H, где произошли ошибки.
3. Синдромы взаимно однозначно соответствуют смежным классам.

$$y_1 + C = y_2 + C \Leftrightarrow y_1 - y_2 \in C \Leftrightarrow H(y_1 - y_2) = 0 \Leftrightarrow Hy_1 = Hy_2$$

Синдромы

Пусть H — проверочная матрица, y — вектор.
Синдром вектора y — это вектор

$$S_y = Hy.$$

Свойства синдрома:

1. $S_x = 0 \iff x \in C$
2. S_y — сумма тех столбцов матрицы H, где произошли ошибки.
3. Синдромы взаимно однозначно соответствуют смежным классам.

Таким образом, чтобы узнать смежный класс, в котором находится принятое слово y (и вектор ошибок e), нужно вычислить S_y. Это быстрее, чем поиск по таблице.

Вероятность ошибки декодирования

Зафиксируем схему декодирования.

Определение. *Вероятность ошибки декодирования* $P_{\text{ош}}$ — вероятность появления на выходе декодера некодового слова.

Вероятность ошибки декодирования

Зафиксируем схему декодирования.

Определение. *Вероятность ошибки декодирования* $P_{\text{ош}}$ — вероятность появления на выходе декодера некодового слова.

Пусть имеем линейный код мощности $M = 2^k$ с кодовыми словами $x_1, \ldots, x_M$, где первые k символов $x_1^i, \ldots, x_k^i$ в каждом слове являются информационными. Пусть $y = (y_1, \ldots, y_n)$ — слово на выходе декодера.

Вероятность ошибки декодирования

Зафиксируем схему декодирования.

Определение. *Вероятность ошибки декодирования* $P_{\text{ош}}$ — вероятность появления на выходе декодера некодового слова.

Пусть имеем линейный код мощности $M = 2^k$ с кодовыми словами $x_1, \ldots, x_M$, где первые k символов $x_1^i, \ldots, x_k^i$ в каждом слове являются информационными. Пусть $y = (y_1, \ldots, y_n)$ — слово на выходе декодера.

Определение. *Вероятность ошибки на символ* $P_{\text{симв}}$ — средняя вероятность того, что после декодирования информационный символ является ошибочным:

Вероятность ошибки декодирования

Зафиксируем схему декодирования.

Определение. *Вероятность ошибки декодирования* $P_{\text{ош}}$ — вероятность появления на выходе декодера некодового слова.

Пусть имеем линейный код мощности $M = 2^k$ с кодовыми словами $x_1, \ldots, x_M$, где первые k символов $x_1^i, \ldots, x_k^i$ в каждом слове являются информационными. Пусть $y = (y_1, \ldots, y_n)$ — слово на выходе декодера.

Определение. *Вероятность ошибки на символ* $P_{\text{симв}}$ — средняя вероятность того, что после декодирования информационный символ является ошибочным:

$$P_{\text{симв}} = \frac{1}{kM} \sum_{j=1}^{k} \sum_{i=1}^{M} P\{y_j \neq x_j^i \mid x^i \text{было послано}\}$$

Теорема Шеннона

Пусть P_C — вероятность ошибки на слово для кода C. За P_i обозначим вероятность неправильного декодирования при условии, что передано слово x^i. Заметим, что P_i зависит от p. Тогда

$$P_C = \frac{1}{M} \sum_{i=1}^{M} P_i.$$

Теорема Шеннона

Пусть P_C — вероятность ошибки на слово для кода C. За P_i обозначим вероятность неправильного декодирования при условии, что передано слово x^i. Заметим, что P_i зависит от p. Тогда

$$P_C = \frac{1}{M} \sum_{i=1}^{M} P_i.$$

Определим величину $P^*(M, n, p)$ как минимум P_C по всем кодам C мощности M при фиксированной вероятности подмены символа p.

$P^*(M, n, p) = \min_{\substack{|C|=M \\ \text{длина}}} P_C$

Теорема Шеннона

Пусть P_C — вероятность ошибки на слово для кода C. За P_i обозначим вероятность неправильного декодирования при условии, что передано слово x^i. Заметим, что P_i зависит от p. Тогда

$$P_C = \frac{1}{M} \sum_{i=1}^{M} P_i.$$

Определим величину $P^*(M, n, p)$ как минимум P_C по всем кодам C мощности M при фиксированной вероятности подмены символа p.

Определим функции **энтропии**:

$$\mathcal{H}(x) = -x \log x - (1 - x) \log(1 - x),$$

Теорема Шеннона

Пусть P_C — вероятность ошибки на слово для кода C. За P_i обозначим вероятность неправильного декодирования при условии, что передано слово x^i. Заметим, что P_i зависит от p. Тогда

$$P_C = \frac{1}{M} \sum_{i=1}^{M} P_i.$$

Определим величину $P^*(M, n, p)$ как минимум P_C по всем кодам C мощности M при фиксированной вероятности подмены символа p.

Определим функции **энтропии**:

$$\mathcal{H}(x) = -x \log x - (1 - x) \log(1 - x),$$

и **пропускной способности**:

$$C(p) = 1 - \mathcal{H}(p) = 1 + x \log x + (1 - x) \log(1 - x).$$

Теорема Шеннона. Пусть R — любое число т.ч. $0 < R < C(p)$. Пусть $M_n = 2^{[n \cdot R]}$. Тогда

$$P^*(M, n, R) \to 0, \quad n \to \infty.$$

Теорема Шеннона. Пусть R — любое число т.ч. $0 < R < C(p)$. Пусть $M_n = 2^{[n \cdot R]}$. Тогда

$$P^*(M, n, R) \to 0, \quad n \to \infty.$$

Другими словами, для достаточно больших n существует хороший код длины n, со скоростью, сколь угодно близкой к пропускной способности канала связи.
Скорость канала определяется как величина $\frac{\log M}{n}$.