Тест по прошлой лекции

https://forms.gle/5jzj1SvgZ2pkW3oG7

Санкт-Петербургский государственный университет DSCS.PRO TICS LAB

Введение в теорию информации

Понятие информации и энтропии

Направление «Искусственный интеллект и наука о данных», 23.Б16-мм, 23.Б18-мм

06.10.2023

Абрамов Максим Викторович, Есин Максим Сергеевич

Понятие вероятности

определение вероятности

вероятность — это отношение благоприятных событий ко всем возможным событиям в одном испытании

формула работает для равновероятных событий

НО

формула вероятности

$$P = \frac{m}{n}$$

m — благоприятные

n — все

свойства вероятностей

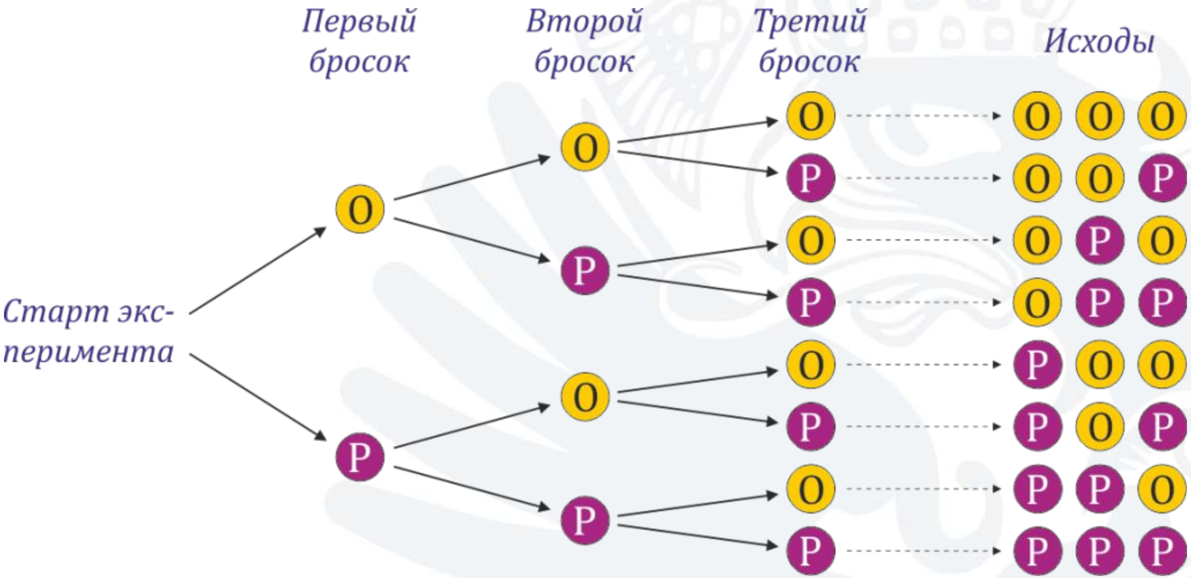
рисунок	свойство	формула
P	вероятность всех возможных событий в испытании равна 1	$P = \frac{n}{n} = 1$
P	вероятность — это число в промежутке от 0 до 1	$0 \leq P \leq 1$
P[−A] P[A]	обратные события — такие, когда в испытании происходит только одно из них	$P[-A] + P[A] = 1$

Правильная монета и кубик

- Правильная монета имеет две отличающиеся стороны – орёл и решка.
- При этом вероятность выпадения одной из сторон – 1/2

- Шестигранный кубик имеет 6 сторон, на каждой из которых – одно из чисел от 1 до 6
- Вероятность выпадения одной из сторон – 1/6
- Вероятность выпадения стороны с четным числом – 3/6

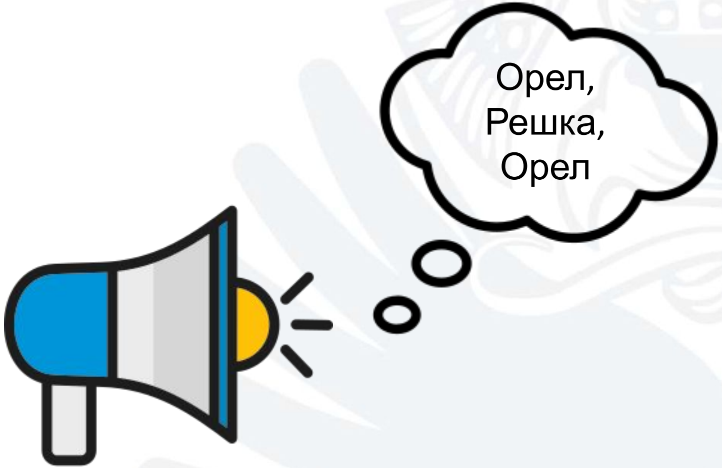
Эксперимент, пространство исходов



Пространство исходов для эксперимента «Три броска монеты»

- Эксперимент состоит в подбрасывании монеты с орлом и решкой трижды
- Исход – результат эксперимента. Для подбрасывания монетки трижды исходом может быть любая из 8 комбинаций

Источник информации



- Источник, сообщающий исход эксперимента, возвращает одно из 8 сообщений с одинаковой вероятностью, равной 1/8
- Как оценить, сколько информации было передано?

Формула Хартли

данное число

логарифм (показатель степени)

$$\log_a N = x$$

основание

$$H = \log_2(N)$$

- *Логарифм числа* N — степень, в которую нужно возвести основание логарифма a, чтобы получить данное число N
- Чтобы оценить количество информации (H) в битовом сообщении длины N, используем *формулу Хартли*

Практические примеры

- Правильную монетку подбросили 7 раз, какое количество информации несет каждый из исходов эксперимента?

- В библиотеке 4 стеллажа, в каждом стеллаже 8 полок. Какое количество информации несёт сообщение о том, что нужная книга находится на пятой полке?

Практические примеры

- Правильную монетку подбросили 7 раз, какое количество информации несет каждый из исходов эксперимента?
- Ответ: **7 бит**

- В библиотеке 4 стеллажа, в каждом стеллаже 8 полок. Какое количество информации несёт сообщение о том, что нужная книга находится на пятой полке?
- Ответ: **3 бита**

Развиваем мысль

- Представим ситуацию: вы ищете пункт выдачи, в котором вам нужно забрать посылку, звучит фраза

«*В том доме живет хотя бы один Иван*»

- В зависимости от того, *в каком городе* фраза была сказана, **она несет разный объем информации о доме**.
    - **Если в одном из районов Москвы** — то **почти никакой информации**, ведь Иванов в Москве много. Вероятность того, что в доме живет хотя бы один Иван, очень велика, поэтому эта фраза никак не сужает границы поиска
    - **Если в одном из районов Пекина** — то **информации в сообщении гораздо больше**: Иванов в Пекине не очень много, поэтому вероятность встретить его в одном из китайских домов очень низкая. Эта фраза сразу вносит определенность и сужает границы поиска

Одинаковая фраза, сказанная в разном контексте, несет в себе разное количество информации

Неравномерное распределение

- Представим ситуацию, когда вероятность некоторых исходов выше других. Пусть эксперимент выглядит так:

«*Вы смотрите за окно и видите машину. Какого она цвета?*»

- Предположим, что вероятность увидеть черную машину равна 1/2, серую — 1/4, белую — 1/8, зеленую — 1/16, а фиолетовую — 1/16. Сколько информации содержится в каждом исходе?
- Формулу Хартли использовать нельзя, потому что исходы не равновероятны
- Более того, разные исходы несут в себе разное количество информации

Энтропия Шеннона

Санкт-Петербургский государственный университет

- Клод Шеннон обобщил формулу Хартли на случай неравновероятных исходов, выдвинув следующие требования к *мере* количества информации – *энтропии*
  - **Непрерывность**: мера должна быть непрерывной — то есть, изменение значения одной из вероятностей на очень маленькую величину должно изменить энтропию также на маленькую величину
  - **Симметрия**: мера не должна меняться, если исходы переупорядочены
  - **Максимум**: мера должна быть максимальной, если все исходы одинаково вероятны (неопределённость является самой высокой, когда все возможные события равновероятны). Для равновероятных событий энтропия должна увеличиваться с увеличением их числа
  - **Аддитивность**: необходима возможность сделать выбор в два шага, в которых значение функции конечного результата должно являться суммой функций промежуточных результатов

$$\underbrace{H\left(\frac{1}{n}, \ldots, \frac{1}{n}\right)}_{n} = H\left(\frac{b_1}{n}, \ldots, \frac{b_k}{n}\right) + \sum_{i=1}^{k} \frac{b_i}{n} \underbrace{H\left(\frac{1}{b_i}, \ldots, \frac{1}{b_i}\right)}_{b_i}.$$

Энтропия Шеннона

- Формально было доказано, что этим требованиям удовлетворяет единственная функция:

$$H(A) = -\sum_{i=1}^{m} P_i \log_2 P_i$$

где A — эксперимент (его еще называют *событием*), P — вероятность i-ого исхода, m — число исходов в пространстве исходов

- Неформально говоря, **энтропия** эксперимента — **мера неопределенности результатов эксперимента**
- Энтропия показывает **среднее число бит в сообщении источника** с неравномерно распределенными вероятностями исходов
- Величина ($-\log_2 P_i$) характеризует **частную энтропию** i-ого источника

Энтропия Шеннона

Санкт-Петербургский государственный университет

- Вернемся к примеру с машинами четырех цветов: вероятность увидеть черную машину равна 1/2, серую — 1/4, белую — 1/8, зеленую — 1/16, а фиолетовую — 1/16. Используя формулу для энтропии Шеннона, **получим энтропию, равную 7/4**
- Представим, что вероятности всех исходов в этом примере равны и пересчитаем энтропию эксперимента. Используя ту же формулу энтропии, **получим 2**
- Неопределенность результатов ожидаемо **выросла**, когда мы внесли изменения в пространстве исходов

А почему информации действительно меньше?

- На первый взгляд кажется, что и в случае равномерного распределения, и в случае неравномерного распределения информация об исходах кодируется одним образом:
    - черная машина — 00
    - серая машина — 01
    - зеленая машина — 10
    - фиолетовая машина — 11
- Тогда почему мы говорим, что часть исходов содержит больше информации, а часть — меньше? Можно ли предложить **способ кодирования исходов**, при котором передаваемая информация кодируется согласно этому принципу?

Санкт-Петербургский государственный университет

НЕРАВНОМЕРНОЕ КОДИРОВАНИЕ

- это кодирование, при котором разные символы могут кодироваться кодами разной длины.

пример:

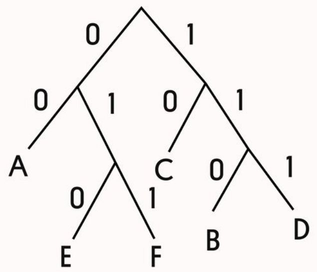
построим коды для 6 объектов: a b c d e f, присвоим им двоичные коды

А	Б	В	Г	Д
000	10	01	110	001

условие фано означает: ни одно кодовое слово не является началом другого кодового слова.

- Неравномерный код позволяет обойтись **минимальным числом битов при кодировании**

как строить дерево?



✦ каждая правая ветка — 1, а левая — 0.

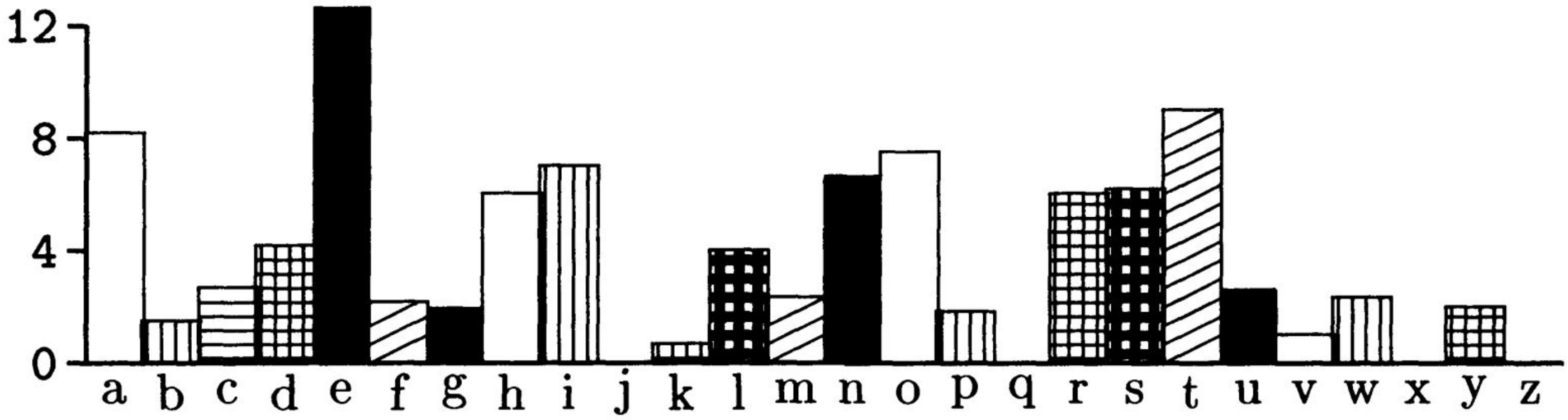
✦ если ветка занята символом, то дальше строить дерево по этой ветке нельзя. это и есть условие Фано.

✦ значение считывается от корня дерева к символу. считать нужно все ветки. так а — это 00, а е — 010.

✦ при соблюдении этих правил код, найденный по дереву, не будет нарушать условие фано.

Практический пример: избыточность алфавита

- Представим пространство исходов как буквы английского алфавита, у которых есть некоторая вероятность появления в тексте. Тогда сможем оценить значение энтропии английского языка



Относительная встречаемость букв английского алфавита (в процентах)

Практический пример: избыточность алфавита

- Однако, текст на любом естественном языке содержит определенные *закономерности и правила*, например, самые распространенные триграммы в английском языке: *the, ing, and, her, ere, ent, tha, nth, was, eth*, или правило русского языка — *жи/ши пиши с буквой и*
- Чтобы учесть эти закономерности, энтропию языка принято считать как предел:

$$H_L = \lim_{n \to \infty} \frac{H(P^n)}{n}.$$

- Вычислить этот предел достаточно сложно, некоторые авторы приводят следующие оценки для оценки энтропии английского языка:

$$1,0 \leqslant H_L \leqslant 1,5.$$

Практический пример: избыточность алфавита

- Каждая буква английского языка одновременно:
  - требует 5 битов для представления в компьютере
  - но содержит в себе не более 1,5 битов информации

- Это свидетельствует о том, что естественный язык имеет **высокую степень избыточности**, в чем легко убедиться на следующем предложении, из которого все еще можно извлечь заложенный в нем смысл, хотя там и удалены по две из каждых четырех букв:

*On** up** a t**e t**re **s a **rl **al**d S**w W**te.*

Краткие итоги

- Хотели узнать, **сколько информации** содержит в себе исход испытания
- Для пространства равновероятных исходов используется **формула Хартли**
- Обобщение формулы Хартли — **энтропия Шеннона** — позволяет работать с неравномерным распределением в пространстве исходов. Энтропия — мера неопределенности результатов эксперимента
- **Наибольшим значением энтропии** будет обладать эксперимент с набором равновероятных событий, а **наименьшим значением** — достоверный эксперимент (событие), то есть **эксперимент с единственным исходом**

Санкт-Петербургский государственный университет | DSCS.PRO | TICS LAB

Введение в теорию информации

Хранение информации в памяти

Направление «Искусственный интеллект и наука о данных», 23.Б16-мм, 23.Б18-мм

06.10.2023

Абрамов Максим Викторович, Есин Максим Сергеевич

Биты

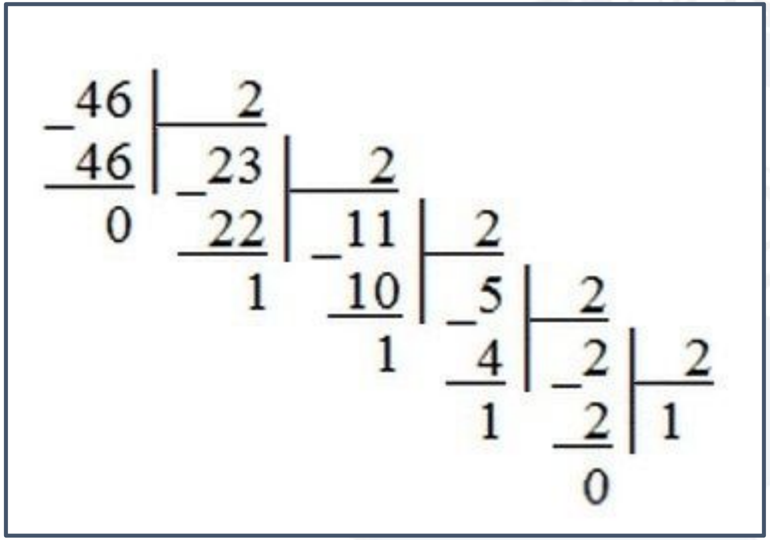
Единицы измерения

1 байт *(byte)*	= **8** бит
1 Кбайт (килобайт)	= **1024** байта
1 Мбайт (мегабайт)	= **1024** Кбайт
1 Гбайт (гигабайт)	= **1024** Мбайт
1 Тбайт (терабайт)	= **1024** Гбайт
1 Пбайт (петабайт)	= **1024** Тбайт

2^{10}

- Бит — это **самая маленькая единица измерения информации**
- Название произошло от слов BInary digiT
- Например, лампочка может передавать один бит информации. Если она включена — это 1, если выключена — 0

Представление данных в памяти



Перевод числа в двоичную систему счисления

- Числа люди привыкли изображать в десятичной системе счисления. Для компьютера удобнее **двоичная система**
- Любые данные (числа, символы, графические и звуковые образы) в компьютере представляются **в виде последовательностей из нулей и единиц**. Значит с ними можно работать как со словами в алфавите {0, 1}
- Такой взгляд роднит **вычислительные машины** с абстрактными вычислителями. Вспомните **машины Тьюринга или нормальные алгоритмы Маркова**

Целые числа без знака

- Для представления 2^k чисел без знака можно использовать битовую последовательность длины k
- С такими последовательностями можно совершать арифметические действия

Число	Двоичный код числа
0	0000 0000
1	0000 0001
2	0000 0010
3	0000 0011
...	...
255	1111 1111

$$\begin{array}{r} 110 \\ \times\ \ 10 \\ \hline 000 \\ +\ 110\ \ \\ \hline 1100 \end{array}$$

$$\begin{array}{r} 1100 \\ +\ \ 101 \\ \hline 10001 \end{array}$$

Целые числа со знаком

- Для представления знаковых целых чисел используются три способа:
  a. **прямой** код
  b. **обратный** код
  c. **дополнительный** код

- Все три способа используют самый **левый (старший) разряд** битового набора длины k **для кодирования знака числа**: знак «плюс» кодируется нулем, а «минус» — единицей

- **Остальные k-1 разрядов** (называемые мантиссой или цифровой частью) используются **для представления абсолютной величины числа**

Представление положительных целых чисел в памяти

Отрицательные целые числа

Прямой код числа –1

1	0	0	0	0	0	0	1

Знак числа "–"

Прямой код числа –127

1	1	1	1	1	1	1	1

Знак числа "–"

Число: –1

Код модуля числа:

0	0	0	0	0	0	0	1

Обратный код числа:

1	1	1	1	1	1	1	0

Знак числа "–"

Число: –127

Код модуля числа:

0	1	1	1	1	1	1	1

Обратный код числа:

1	0	0	0	0	0	0	0

Знак числа "–"

Дополнительный код

Число: –1

Код модуля числа:

0	0	0	0	0	0	0	1

Обратный код числа:

1	1	1	1	1	1	1	0

Знак числа "–"

Число: –127

Код модуля числа:

0	1	1	1	1	1	1	1

Обратный код числа:

1	0	0	0	0	0	0	0

Знак числа "–"

Дополнительный код числа –1:

1	1	1	1	1	1	1	1

Знак числа "–"

Дополнительный код числа –127:

1	0	0	0	0	0	0	1

Знак числа "–"

Интерпретации чисел

Санкт-Петербургский государственный университет

Ниже приведена таблица, демонстрирующая различные интерпретации битовых наборов длины 3.

Что представляет / Битовый набор (k =3)	Беззнаковое целое	Знаковое целое в прямом коде	Знаковое целое в обратном коде	Знаковое целое в дополнительном коде
000	0	+0	+0	+0
001	1	+1	+1	+1
010	2	+2	+2	+2
011	3	+3	+3	+3
100	4	−0	−3	−4
101	5	−1	−2	−3
110	6	−2	−1	−2
111	7	−3	−0	−1

Зачем вообще нужен обратный код?

- Дополнительный код не только позволяет заменить операцию вычитания на операцию сложения.
- Он позволяет сделать операции сложения и вычитания одинаковыми для знаковых и беззнаковых чисел
- Это упрощает упрощает архитектуру ЭВМ