

# Кластеризация данных

---

Александр Котов, гр. 5538

Николай Красильников, гр. 5538

# План доклада

---

- 1. Основные определения**
2. Общая схема кластеризации
3. Популярные алгоритмы
4. Применения кластеризации

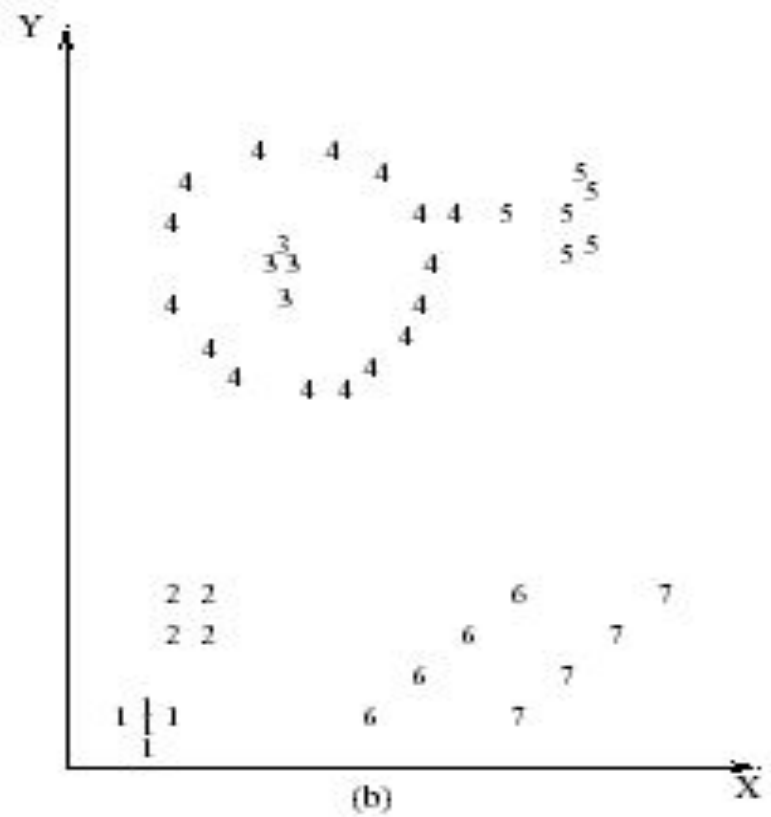
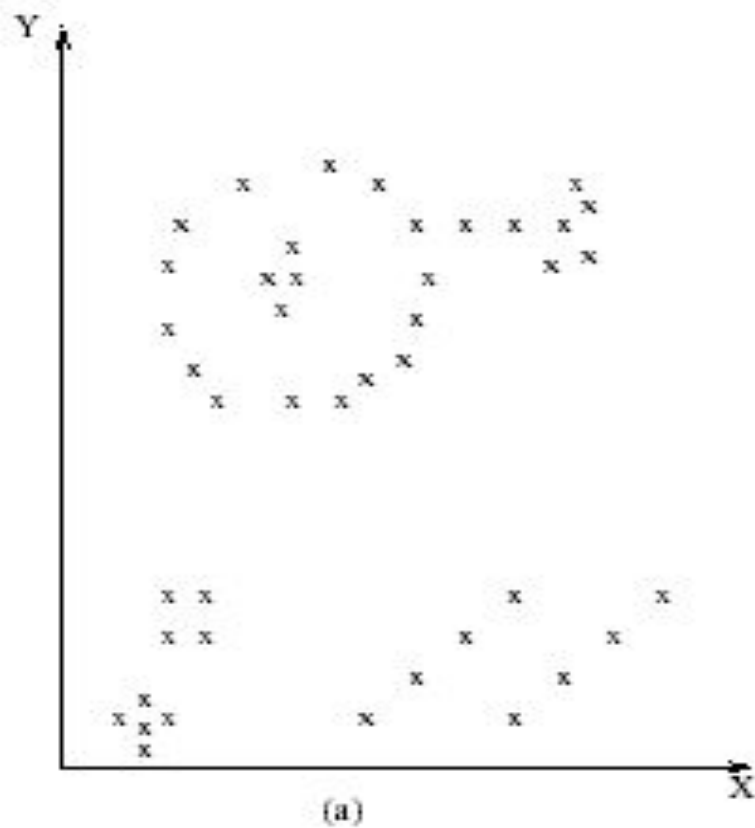
# Что такое кластеризация?

---

- **Кластеризация** – это автоматическое разбиение элементов некоторого множества (объекты, данные, вектора характеристик) на группы (кластеры) по принципу схожести.

# Кластеризация (пример)

---



# Разница между кластеризацией и классификацией

---

- Кластеризация (unsupervised classification) разбивает множество объектов на группы, которые определяются только ее результатом.
- Классификация (supervised classification) относит каждый объект к одной из заранее определенных групп.

# Зачем нужна кластеризация?

---

- Много практических применений в информатике и других областях:
  - Анализ данных (Data mining);
  - Группировка и распознавание объектов;
  - Извлечение и поиск информации.
- Это важная форма абстракции данных.
- Это активно развивающаяся область теоретической информатики.

# Формальные определения

---

- Вектор характеристик (объект)  $\mathbf{x}$  – единица данных для алгоритма кластеризации. Обычно это элемент  $d$ -мерного пространства:  
 $\mathbf{x} = (x_1, \dots, x_d)$ .
- Характеристика (атрибут)  $x_i$  – скалярная компонента вектора  $\mathbf{x}$ .
- Размерность  $d$  – количество характеристик объекта  $\mathbf{x}$ .

# Формальные определения (продолжение)

---

- Множество объектов  $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_n\}$  – набор входных данных.  $i$ -й объект из  $\mathbf{X}$  определяется как  $\mathbf{x}_i = (x_{i,1}, \dots, x_{i,d})$ . Часто  $\mathbf{X}$  представляют в виде *матрицы характеристик* размера  $n \times d$ .
- Кластер – подмножество «близких друг к другу» объектов из  $\mathbf{X}$ .
- Расстояние  $d(\mathbf{x}_i, \mathbf{x}_j)$  между объектами  $\mathbf{x}_i$  и  $\mathbf{x}_j$  – результат применения выбранной метрики (или квази-метрики) в пространстве характеристик.



# Постановка задачи

---

- Цель кластеризации – построить оптимальное разбиение объектов на группы:
  - разбить  $N$  объектов на  $k$  кластеров;
  - просто разбить  $N$  объектов на кластеры.
- Оптимальность может быть определена как требование минимизации среднеквадратической ошибки разбиения:

$$e^2(X, L) = \sum_{j=1}^K \sum_{i=1}^{n_j} \|x_i^{(j)} - c_j\|^2$$

# План доклада

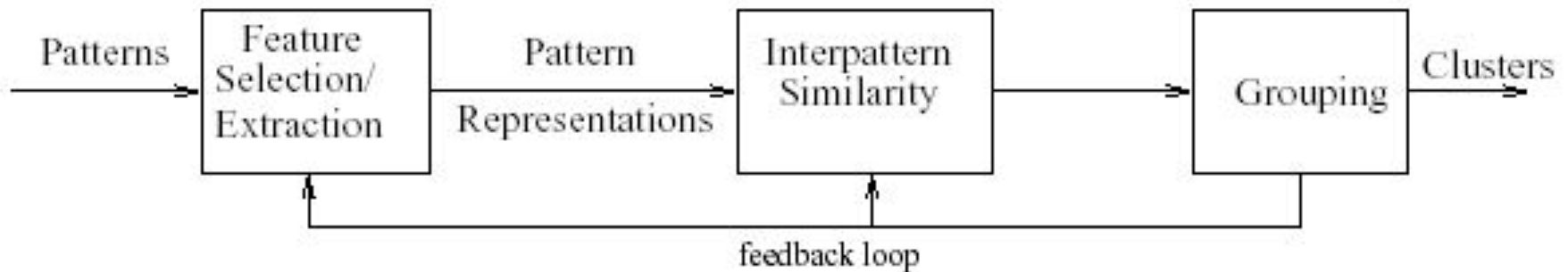
---

1. Основные определения
2. **Общая схема кластеризации**
3. Популярные алгоритмы
4. Применения кластеризации

# Общая схема кластеризации

---

1. Выделение характеристик
2. Определение метрики
3. Разбиение объектов на группы
4. Представление результатов



# Выделение характеристик

---

1. Выбор свойств, характеризующих объекты:
  - количественные характеристики (координаты, интервалы...);
  - качественные характеристики (цвет, статус, воинское звание...).
2. Уменьшение размерности пространства, нормализация характеристик.
3. Представление объектов в виде характеристических векторов.

# Выбор метрики

---

- Метрика выбирается в зависимости от:
  - пространства, где расположены объекты;
  - неявных характеристик кластеров.
- Если все координаты объекта непрерывны и вещественны, а кластеры должны представлять собой нечто вроде гиперсфер, то используется классическая метрика Евклида (на самом деле, чаще всего так и есть):

$$d_2(x_i, x_j) = \left( \sum_{k=1}^d (x_{i,k} - x_{j,k})^2 \right)^{1/2} = \|x_i - x_j\|_2$$

# План доклада

---

1. Основные определения
2. Общая схема кластеризации
3. **Популярные алгоритмы**
4. Применения кластеризации

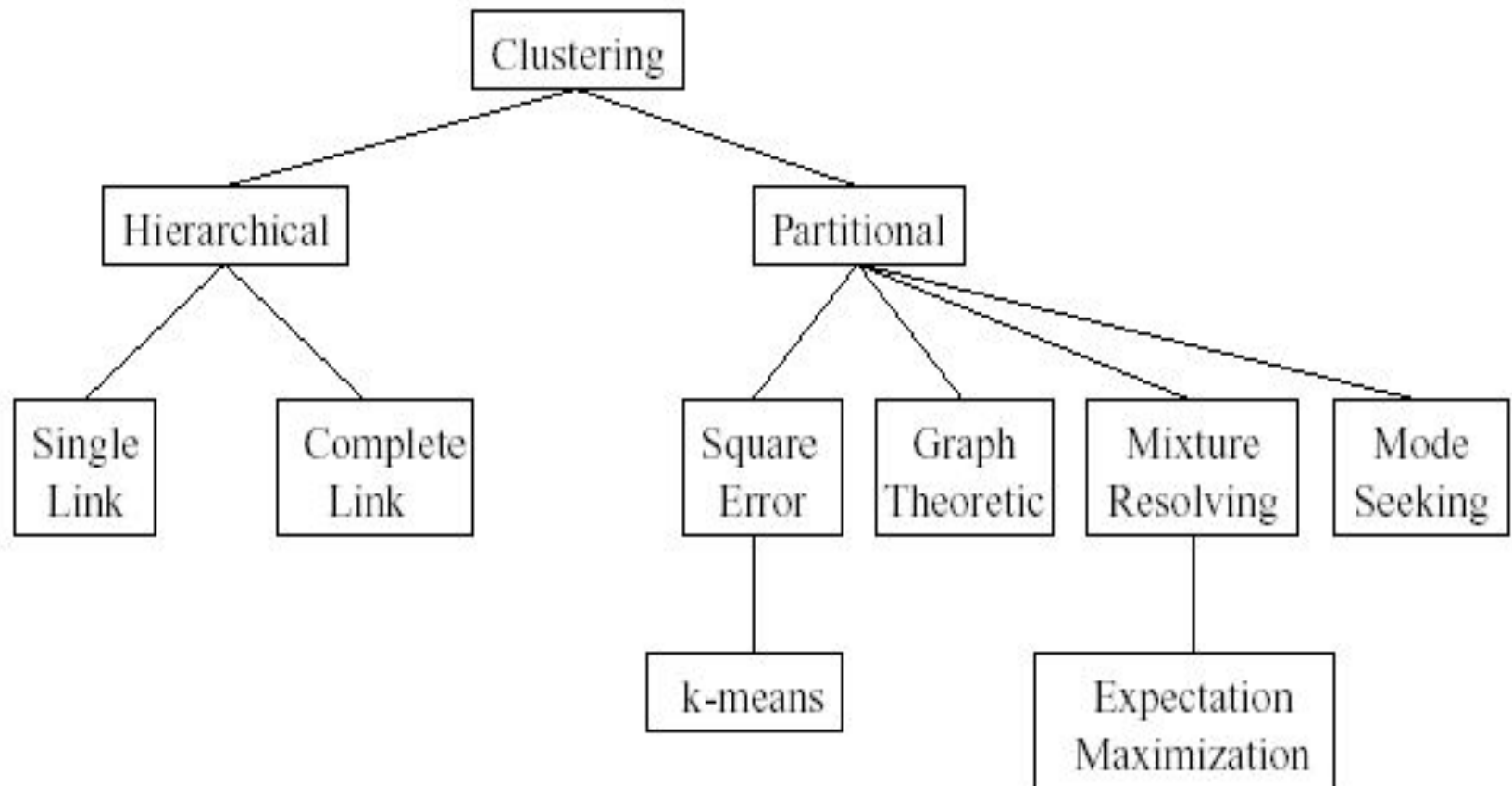
# Алгоритмы кластеризации

---

- Иерархические алгоритмы
- Минимальное покрывающее дерево
- *k*-Means алгоритм (алгоритм *k*-средних)
- Метод ближайшего соседа
- Алгоритмы нечеткой кластеризации
- Применение нейронных сетей
- Генетические алгоритмы
- Метод закалки

# Алгоритмы кластеризации (схема)

---





# Классификация алгоритмов

---

- Строящие «снизу-вверх» и «сверху-вниз»
- Монотетические и политетические
- Непересекающиеся и нечеткие
- Детерминированные и стохастические
- Поточковые (online) и не поточковые
- Зависящие и не зависящие от начального разбиения
- Зависящие и не зависящие от порядка рассмотрения объектов

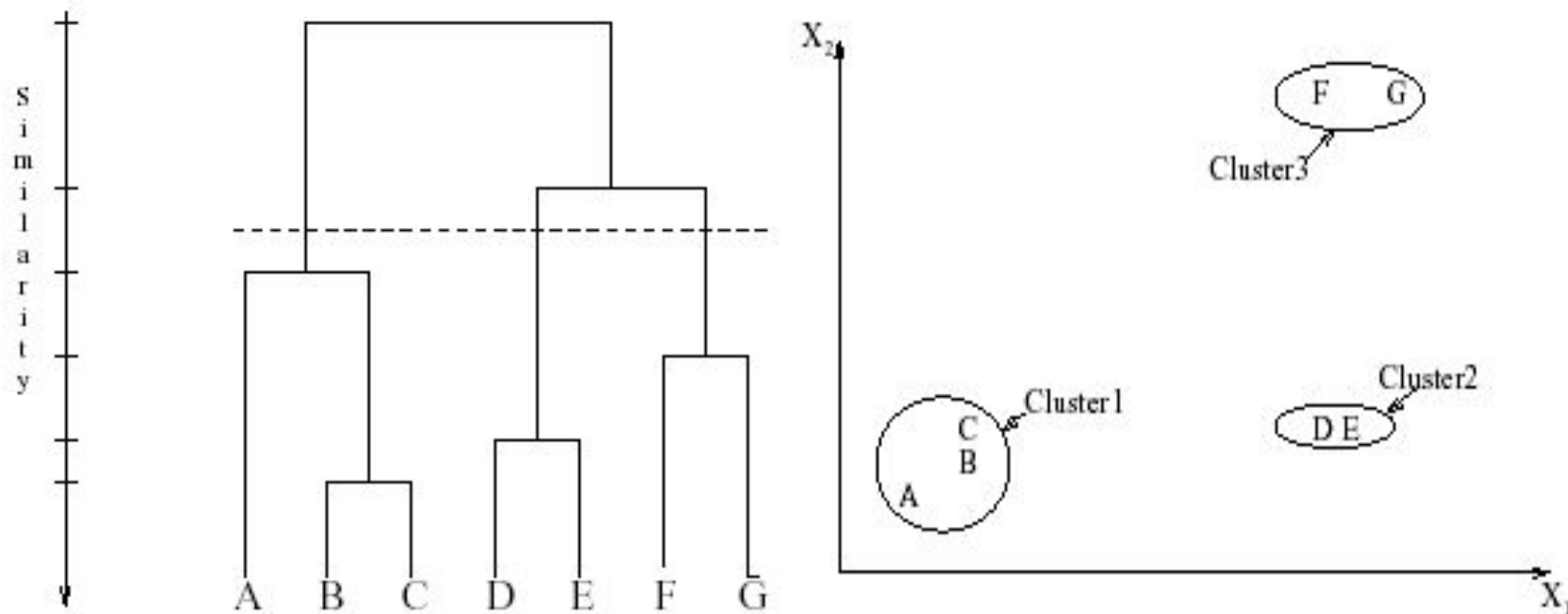
# Иерархические алгоритмы

---

- Результатом работы является *дендограмма* (иерархия), позволяющая разбить исходное множество объектов на любое число кластеров.
- Два наиболее популярных алгоритма, оба строят разбиение «снизу-вверх»:
  - Single-link – на каждом шаге объединяет два кластера с наименьшим расстоянием между двумя *наиболее близкими* представителями;
  - Complete-link – объединяет кластеры с наименьшим расстоянием между двумя *наиболее удаленными* представителями.

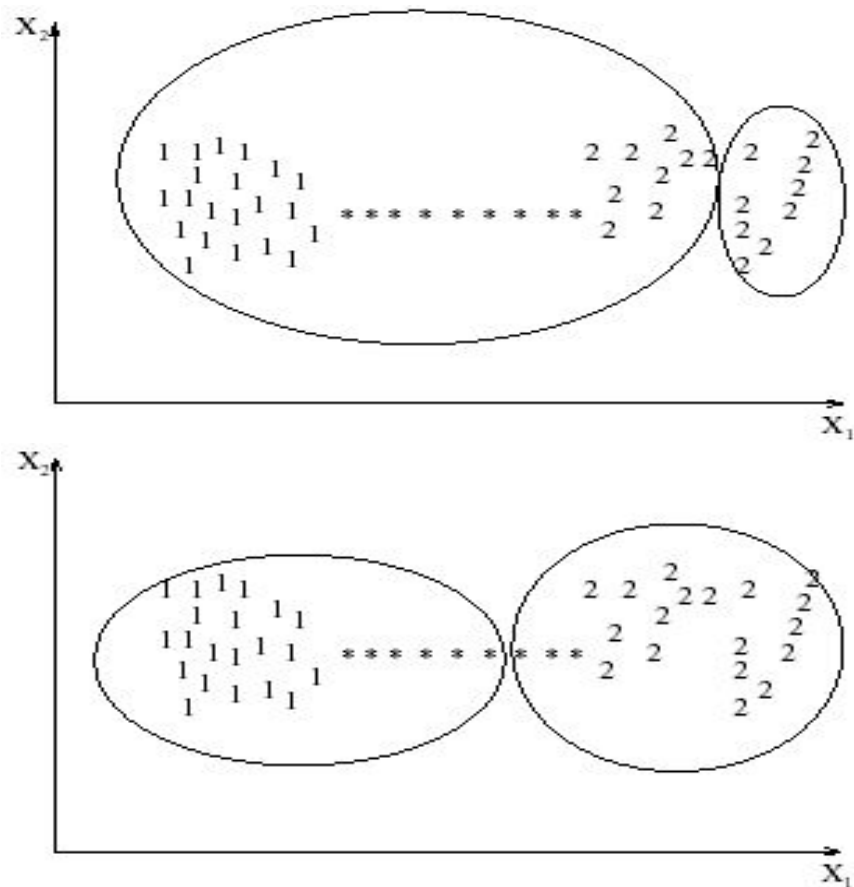
# Single-link (пример)

---



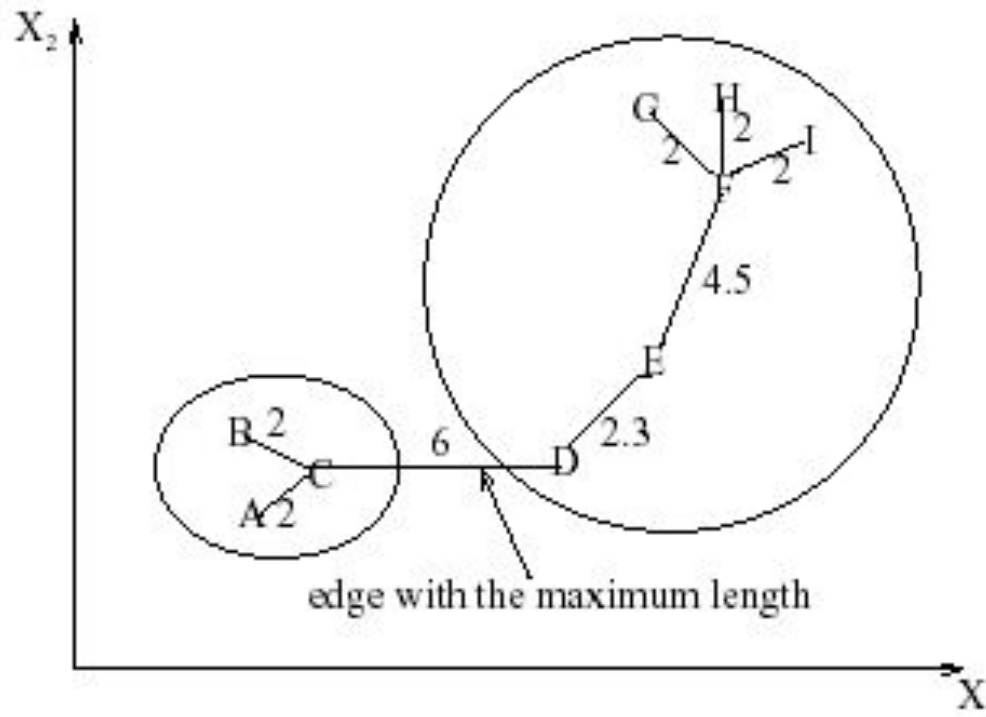
# Сравнение Single-link и Complete-link

---



# Минимальное покрывающее дерево

- Позволяет производить иерархическую кластеризацию «сверху-вниз»:



# $k$ -Means алгоритм

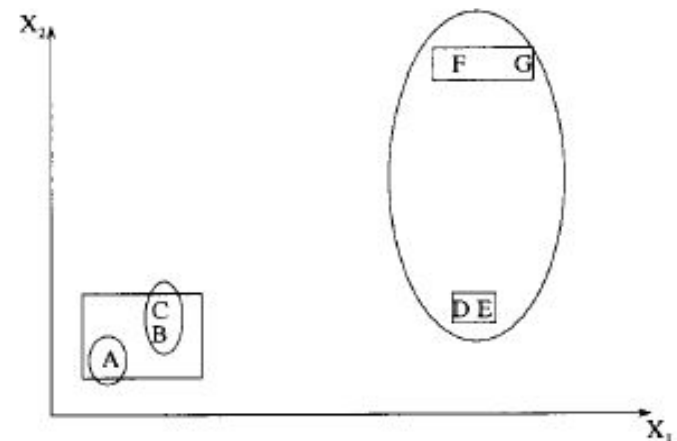
---

1. Случайно выбрать  $k$  точек, являющихся начальными «центрами масс» кластеров (любые  $k$  из  $n$  объектов, или вообще  $k$  случайных точек).
2. Отнести каждый объект к кластеру с ближайшим «центром масс».
3. Пересчитать «центры масс» кластеров согласно текущему членству.
4. Если критерий остановки алгоритма не удовлетворен, вернуться к шагу 2.

# *k*-Means алгоритм (продолжение)

---

- В качестве критерия остановки обычно выбирают один из двух:
  - Отсутствие перехода объектов из кластера в кластер на шаге 2;
  - Минимальное изменение среднеквадратической ошибки.
- Алгоритм чувствителен к начальному выбору «центров масс».



# Метод ближайшего соседа

---

- Один из старейших (1978), простейших и наименее оптимальных алгоритмов:

Пока существуют объекты вне кластеров {

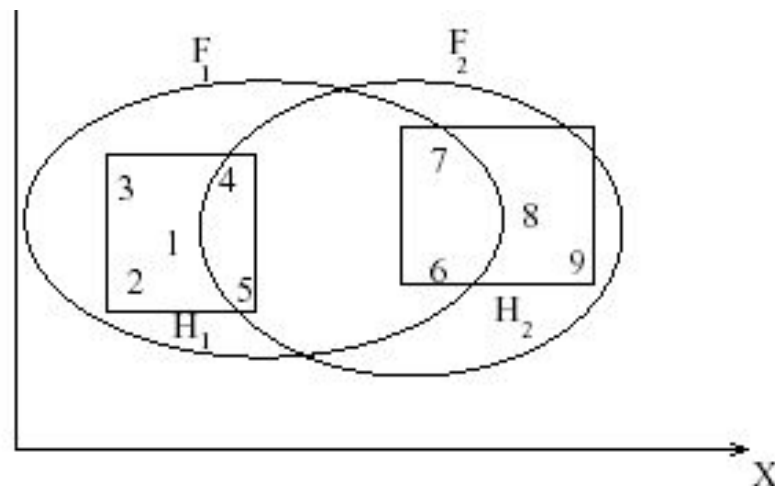
- Для каждого такого объекта выбрать ближайшего соседа, кластер которого определен, и если расстояние до этого соседа меньше порога – отнести его в тот же кластер, иначе можно создать новый;
- Увеличить порог при необходимости;

}



# Нечеткая кластеризация

- Непересекающаяся (четкая) кластеризация относит объект только к одному кластеру.
- Нечеткая кластеризация считает для каждого объекта  $x_i$  степень его принадлежности  $u_{ik}$  к каждому из  $k$  кластеров.



$$F1 = \{(1,0.9), (2,0.8), (3,0.7), (4,0.6), (5,0.55), (6,0.2), (7,0.2), (8,0.0), (9,0.0)\}$$

$$F2 = \{(1,0.0), (2,0.0), (3,0.0), (4,0.1), (5,0.15), (6,0.4), (7,0.35), (8,1.0), (9,0.9)\}$$

# Схема нечеткой кластеризации

---

1. Выбрать начальное нечеткое разбиение  $n$  объектов на  $k$  кластеров путем выбора матрицы принадлежности  $U$  размера  $n \times k$  (обычно  $u_{ik} \in [0,1]$ )

2. Используя матрицу  $U$ , найти значение критерия нечеткой ошибки (например,

$$E^2(X, U) = \sum_{i=1}^N \sum_{k=1}^K u_{ik} \|x_i - c_k\|^2 \quad c_k = \sum_{i=1}^N u_{ik} x_i$$

Перегруппировать объекты с целью ее уменьшения.

3. Пока матрица  $U$  меняется, повторять шаг 2.

# Применение нейронных сетей

---

- Искусственные нейронные сети (ИНС) легко работают в распределенных системах в силу своей природы.
- ИНС могут проводить кластеризацию только для объектов с числовыми атрибутами.
- Настройка весовых коэффициентов ИНС помогает сделать выбор характеристик (этап 1 кластеризации) менее субъективным.
- Кластеризация с применением самоорганизующихся карт Кохонена эквивалентна алгоритму  $k$ -Means.

# Генетические алгоритмы

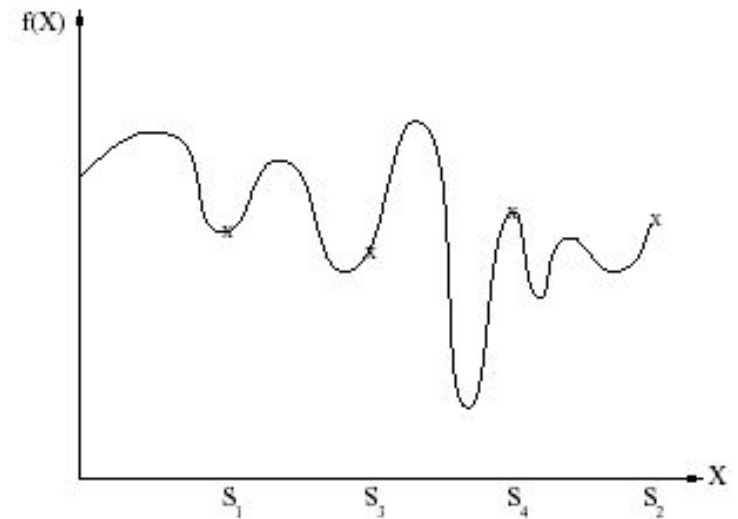
---

1. Выбрать начальную случайную популяцию для множества решений. Получить оценку качества для каждого решения ( $\sim 1 / e^2$ ).
2. Создать и оценить следующую популяцию решений, используя операторы:
  - выбора – предпочитает хорошие решения;
  - рекомбинации («кроссовер») – создает новое решение из двух существующих;
  - мутации – создает новое решение из случайного изменения существующего.
3. Повторять шаг 2 пока это необходимо.

# Генетические алгоритмы ищут глобальный минимум

---

- Большинство популярных алгоритмов оптимизации выбирают начальное решение, которое затем изменяется в ту или иную сторону. Таким образом получается *хорошее* разбиение, но не всегда – *самое оптимальное*.
- Операторы рекомбинации и мутации позволяют получить решения, сильно не похожие на исходные.



# Метод закалки

---

Пытается найти глобальный оптимум, однако работает только с одним текущим решением.

1. Случайно выбрать начальное разбиение  $P_0$  и сосчитать ошибку  $E_{P_0}$ . Выбрать значения начальной и конечной температур ( $T_0 > T_f$ ).
2. Выбрать  $P_1$  недалеко от  $P_0$ . Если  $E_{P_0} > E_{P_1}$ , то утвердить  $P_1$ , иначе –  $P_0$ , но с вероятностью, зависящей от разницы температур. Повторить выбор соседних разбиений несколько раз.
3. Чуть-чуть «остыть»:  $T_0 = c * T_0$ , где  $c < 1$ . Если  $T_0 > T_f$  – снова на шаг 2, иначе – стоп.

# Какой алгоритм выбрать?

---

- Генетические алгоритмы и искусственные нейронные сети хорошо распараллеливаются.
- Генетические алгоритмы и метод закаливания осуществляют глобальный поиск, но метод закаливания сходится о-о-очень медленно.
- Генетические алгоритмы хорошо работают только для одно- (двух-) мерных объектов, зато не требуется непрерывность координат.

# Какой алгоритм выбрать? (продолжение)

---

- $k$ -Means быстро работает и прост в реализации, но создает только кластеры, похожие на гиперсферы.
- Иерархические алгоритмы дают оптимальное разбиение на кластеры, но их трудоемкость квадратична.
- На практике лучше всего зарекомендовали себя гибридные подходы, где шлифовка кластеров выполняется методом  $k$ -Means, а первоначальное разбиение – одним из более сильных методов.



# Априорное использование природы кластеров в алгоритмах

---

- Неявное использование:
  - выбор соответствующих характеристик объектов из всех характеристик
  - выбор метрики (метрика Евклида обычно дает гиперсферические кластеры)
- Явное использование:
  - подсчет схожести (использование  $\infty$  для расстояния между объектами из заведомо разных кластеров)
  - представление результатов (учет явных ограничений)

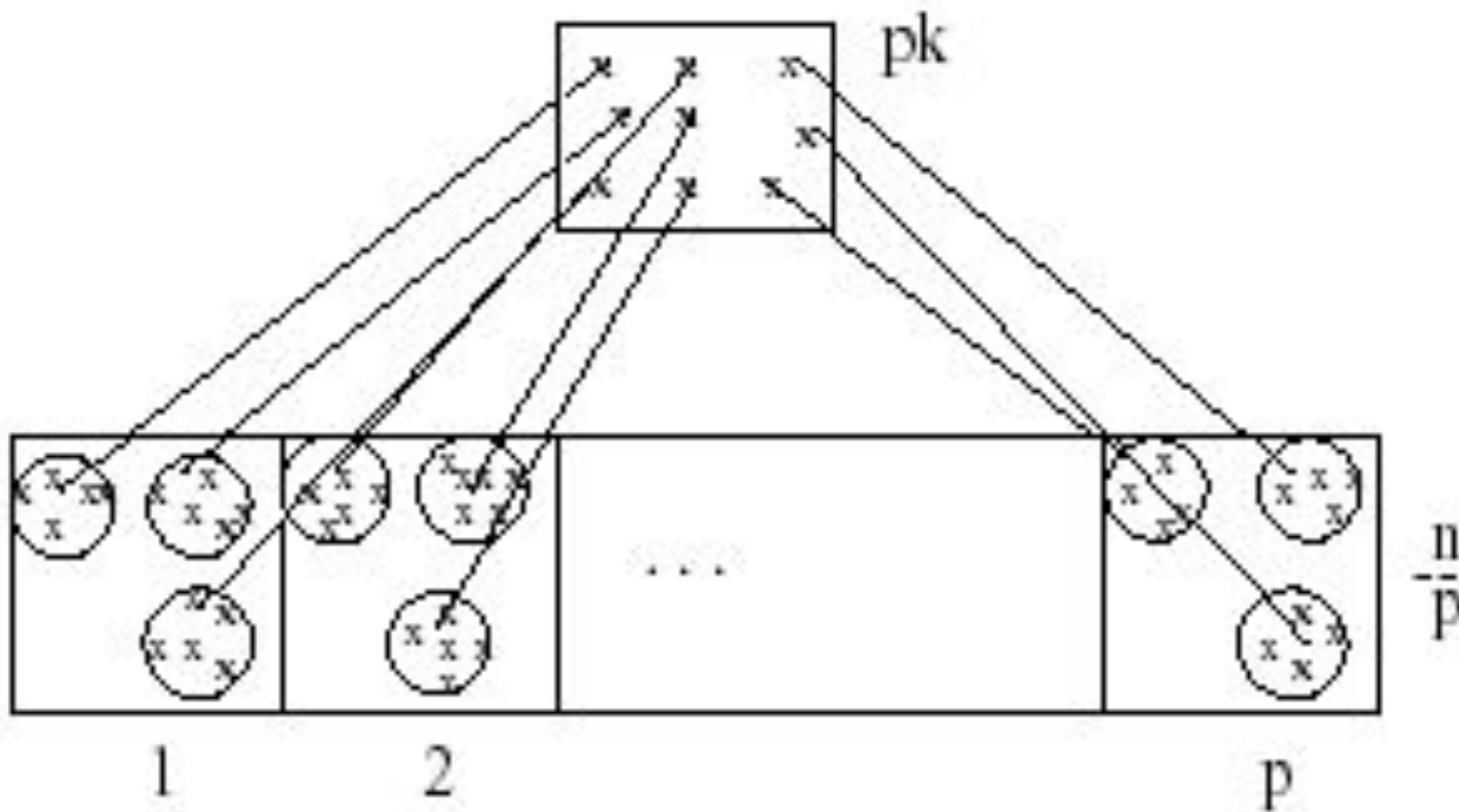
# Кластеризация больших объемов данных

---

- Обычно используют  $k$ -Means или его гибридные модификации.
- Если множество объектов не помещается в основную память, можно:
  - проводить кластеризацию по принципу «разделяй и властвуй»;
  - использовать потоковые (on-line) алгоритмы (например, leader, модификация метода ближайшего соседа);
  - использовать параллельные вычисления.

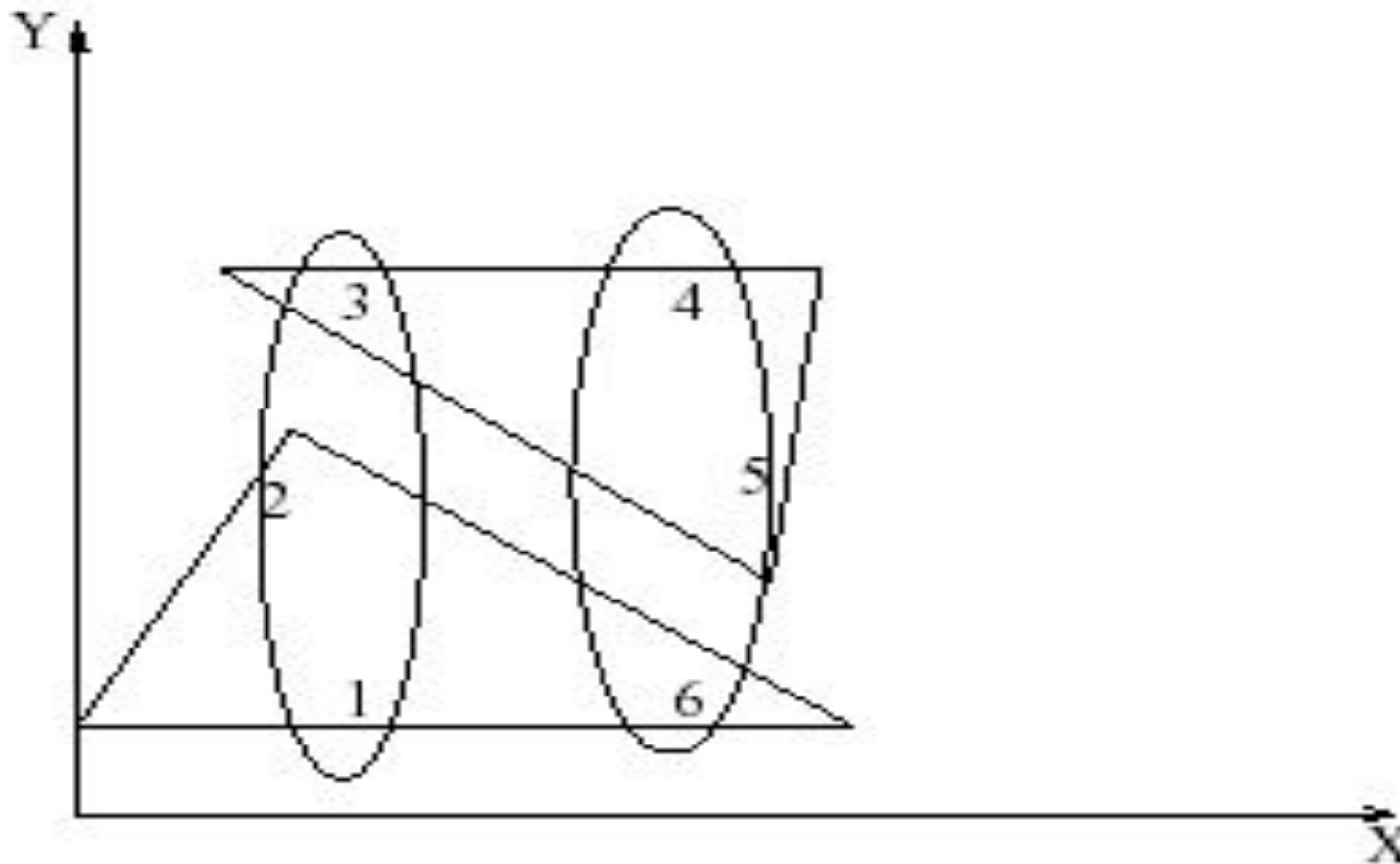
# Разделяй и властвуй (пример)

---



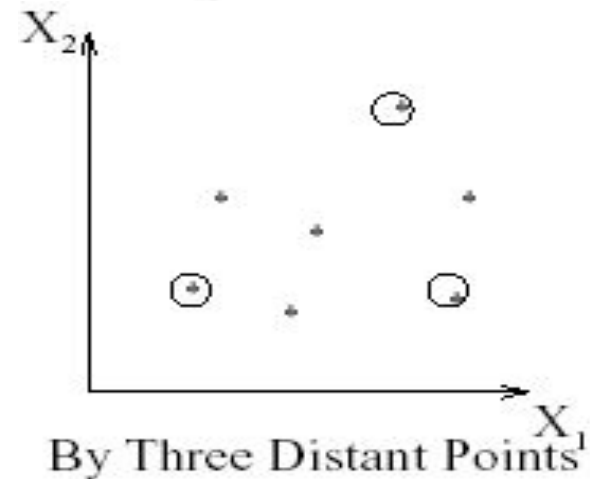
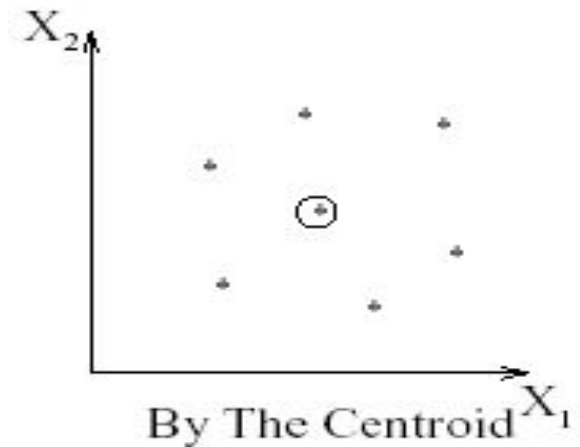
# Алгоритм Leader (пример)

---



# Представление результатов

- Обычно используется один из следующих способов:
  - представление кластеров центроидами;
  - представление кластеров набором характерных точек;
  - представление кластеров их ограничениями.



# План доклада

---

1. Основные определения
2. Общая схема кластеризации
3. Популярные алгоритмы
4. **Применения кластеризации**

# Применения кластеризации

---

- Анализ данных (Data mining)
  - Упрощение работы с информацией
  - Визуализация данных
- Группировка и распознавание объектов
  - Распознавание образов
  - Группировка объектов
- Извлечение и поиск информации
  - Построение удобных классификаторов

# Анализ данных (Data mining)

---

- Упрощение работы с информацией:
  - достаточно работать только с представителями кластеров;
  - легко найти «похожие» объекты – такой поиск применяется в ряде поисковых движков (<http://www.nigma.ru> легко найти «похожие» объекты – такой поиск применяется в ряде поисковых движков (<http://www.nigma.ru>, <http://www.vivisimo.com>, ...));
  - автоматическое построение каталогов.

- Наглядное представление кластеров позволяет понять структуру множества



# <http://www.nigma.ru> (пример)



результаты кластеризации:

**кластеризация [539242]**

- » [использовать](#) (17)
- » [технология кластеризации](#) (16)
- » [алгоритмы кластеризации](#) (15)
- » [серверов](#) (15)
- » [решение](#) (14)
- » [объектов](#) (11)
- » [кластеризация позволяет](#) (10)
- » [метод кластеризации](#) (10)
- » [поиск](#) (10)
- » [возможность](#) (10)
- » [документ](#) (9)
- » [множество](#) (9)
- » [баз данных](#) (7)
- » [введение](#) (7)

кластеризация

искать в:  Google  Yahoo  MSN  Y  
 Altavista  Aport  Nigma

страницы: [1](#) | [2](#) | [3](#) | [4](#) | [5](#) | [6](#) | [7](#) | [8](#) | [9](#) | [10](#) | [11](#) | [12](#) | [13](#) | [14](#) | [15](#) | [1](#)

## Результаты поиска

Найдено примерно : 539 242

### 1. [Кластеризация Windows](#)

**Кластеризация Windows** предоставляет три разных, но до друга, ... Дополнительные сведения о способах сочетания **1 кластеризации Windows** ...

Найти слова | [www.microsoft.com/technet/prodtechnol/windowsserver2003/358b9815-3cd3-4912-a75a-cae85ea8d5ab.mspx](http://www.microsoft.com/technet/prodtechnol/windowsserver2003/358b9815-3cd3-4912-a75a-cae85ea8d5ab.mspx) Google: 1 Google-M: 1

### 2. [Возможности масштабируемости и кластеризации](#)

Технология **кластеризации** в Windows Server 2003 (EN) ... устоявшаяся технология масштабирования компьютерных с применяющаяся ...

Найти слова | [www.microsoft.com/Rus/Business/Infrastructure/Unified/Sc](http://www.microsoft.com/Rus/Business/Infrastructure/Unified/Sc)

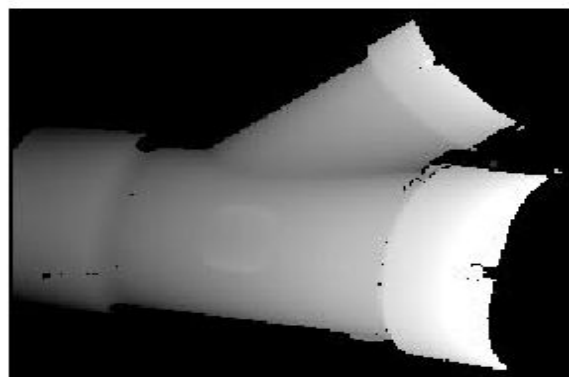
# Группировка и распознавание объектов

---

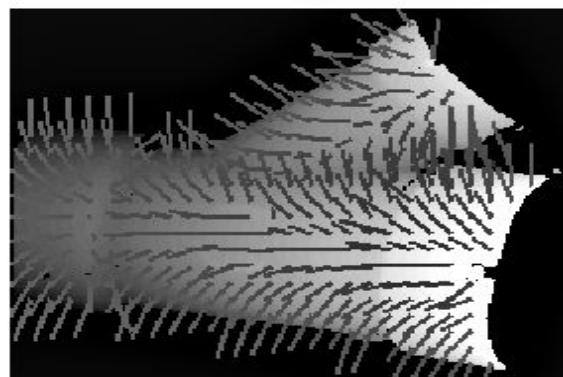
- Распознавание образов (OCR и др.):
  1. построение кластеров на основе большого набора учебных данных;
  2. пометка каждого из кластеров;
  3. ассоциация каждого объекта на входе алгоритма распознавания с меткой соответствующего кластера.
- Группировка объектов:
  - сегментация изображений;
  - уменьшение количества информации.

# Сегментация изображений (пример)

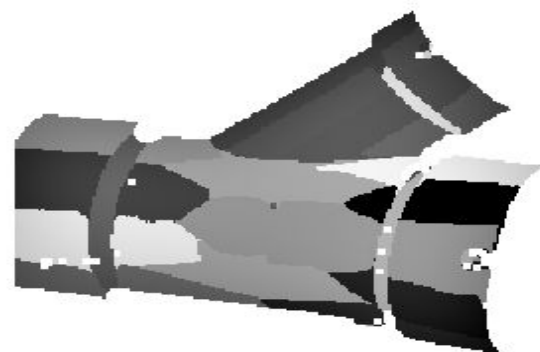
---



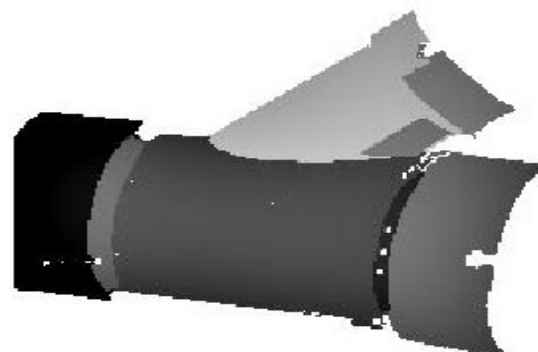
(a)



(b)



(c)



(d)

# Извлечение и поиск информации (на примере книг в библиотеке)

---

- LCC (Library of Congress Classification):
  - Метки с QA76 до QA76.8 – книги по CS.
- Проблемы LCC:
  - книга относится только к одной категории;
  - классификация отстает от быстрого развития некоторых областей науки.
- Выручает автоматическая кластеризация:
  - Нечеткое разбиение на группы решает проблему одной категории;
  - Кластеры вырастают с развитием области.

# Итого

---

- Кластеризация – это автоматическое разбиение множества объектов на группы по принципу схожести
- Общая схема кластеризации одна (выделение характеристик -> выбор метрики -> группировка объектов -> представление результатов). Но существует много различных реализаций этой схемы.
- Кластеризация данных широко применяется в современной информатике.

# Спасибо!

---

## ВОПРОСЫ?