

# Кластеризация

# Повторение

# Задача понижения размерности

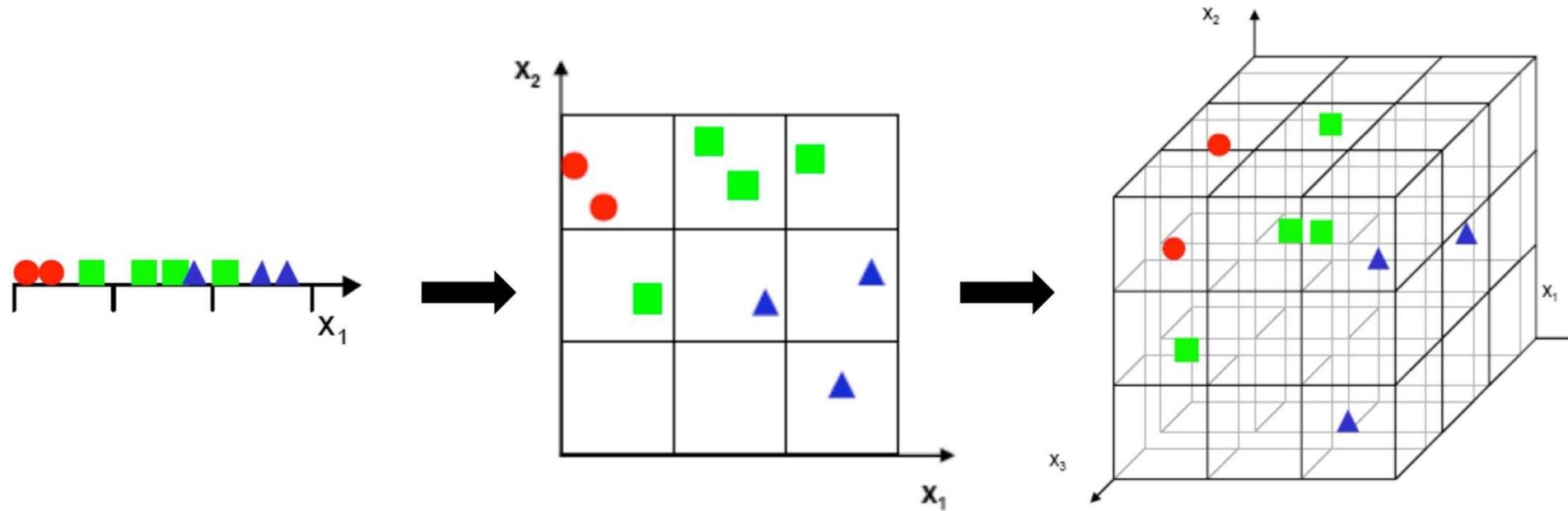
- Дано: матрица «объекты-признаки»  $X$  размера  $\ell \times D$
- Найти: новую матрицу «объекты-признаки»  $Z$  размера  $\ell \times d$
- $d < D$

# Проклятие размерности

- Задача: классификация пончиков на вкусные и невкусные
- 100 объектов
- Цвет: 10 вариантов
- Цвет + размер:  $10 * 4 = 40$  вариантов
- Цвет + размер + форма:  $10 * 4 * 4 = 160$  вариантов



# Проклятие размерности



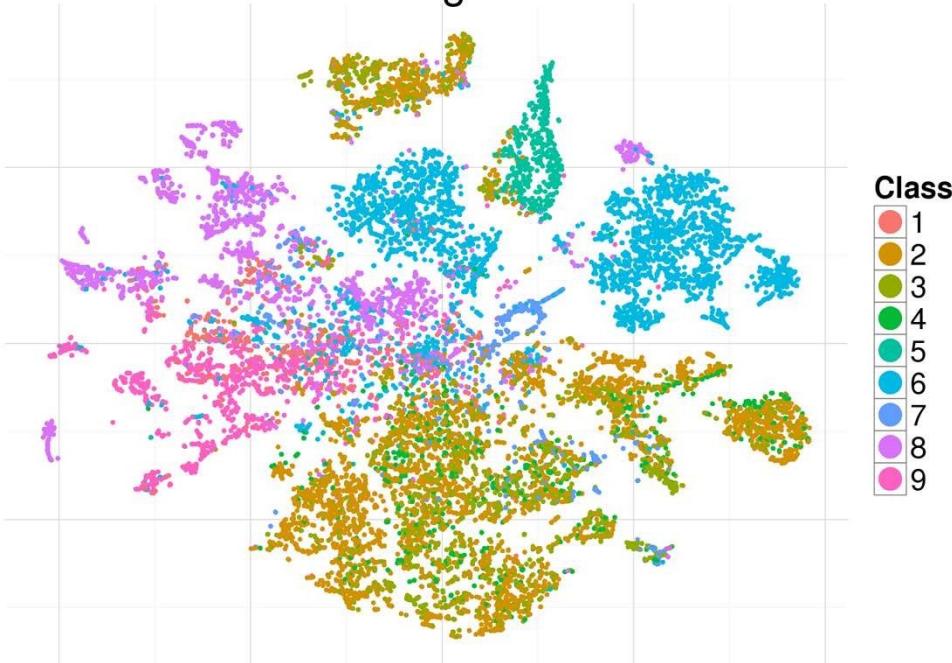
# Ускорение моделей

- Чем больше признаков, тем дольше обучаются модели
- Чем дольше обучаются модели, тем меньше экспериментов удаётся провести
- Чем сложнее модели, тем дольше они вычисляют прогнозы
- Могут быть жёсткие ограничения на скорость
- Пример: рекомендательные системы

# Визуализация



t-SNE 2D Embedding of Products Data



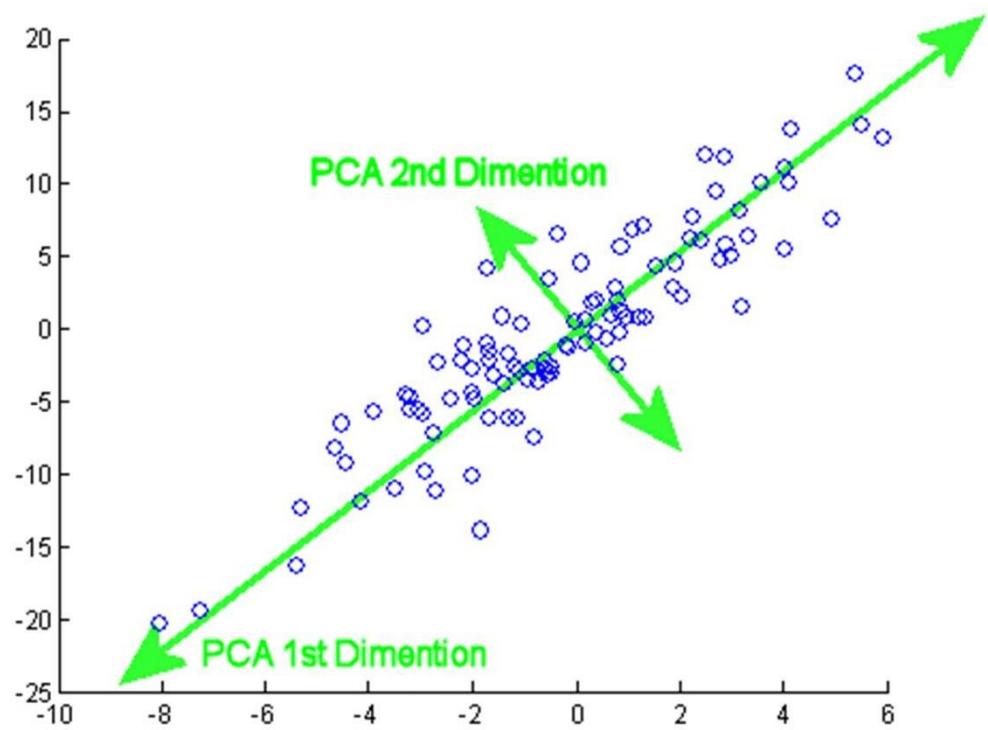
# Методы понижения размерности

- Отбор признаков (feature selection)
  - Выбрать  $d$  самых важных признаков
- Извлечение признаков (feature extraction)
  - Найти  $d$  новых признаков, выражающихся через исходные

# Метод главных компонент

- Principal component analysis (PCA)
- Проецирует данные в пространство меньшей размерности
- Относится к **методам извлечения признаков**

# Извлечение признаков



# Извлечение признаков

- Исходные признаки:  $x_{ik}, D$  штук
- Новые признаки:  $z_{ij}, d$  штук
- Линейный подход:

$$z_{ij} = \sum_{k=1}^D w_{jk} x_{ik}$$

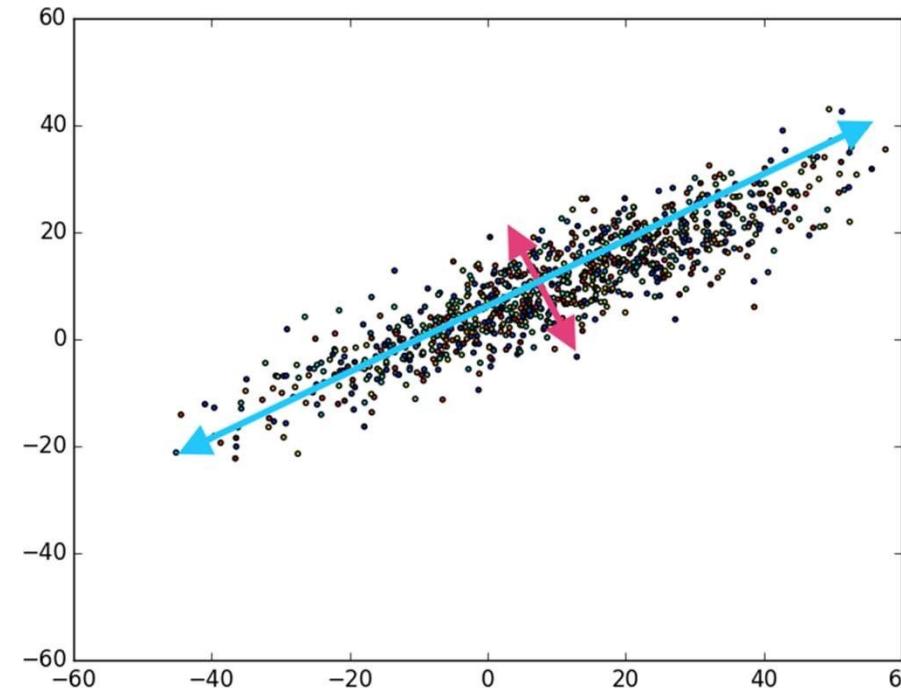
Новые признаки

Вклад исходного  $k$ -го  
признака в новый  $j$ -й

Исходные признаки

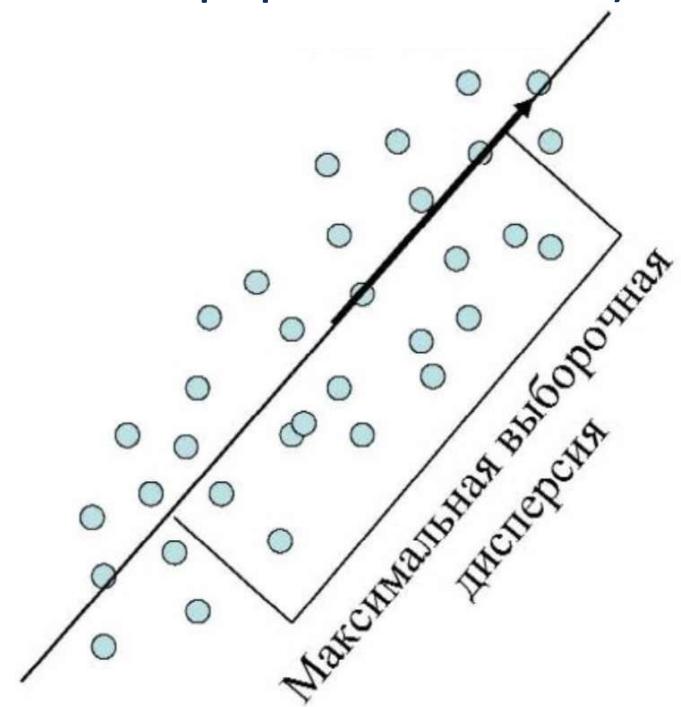
# Метод главных компонент

- Геометрический смысл — поиск гиперплоскости для проецирования выборки
- Как выбирать гиперплоскость?
- Чем выше дисперсия выборки после проецирования, тем лучше
- Дисперсия — мера количества информации



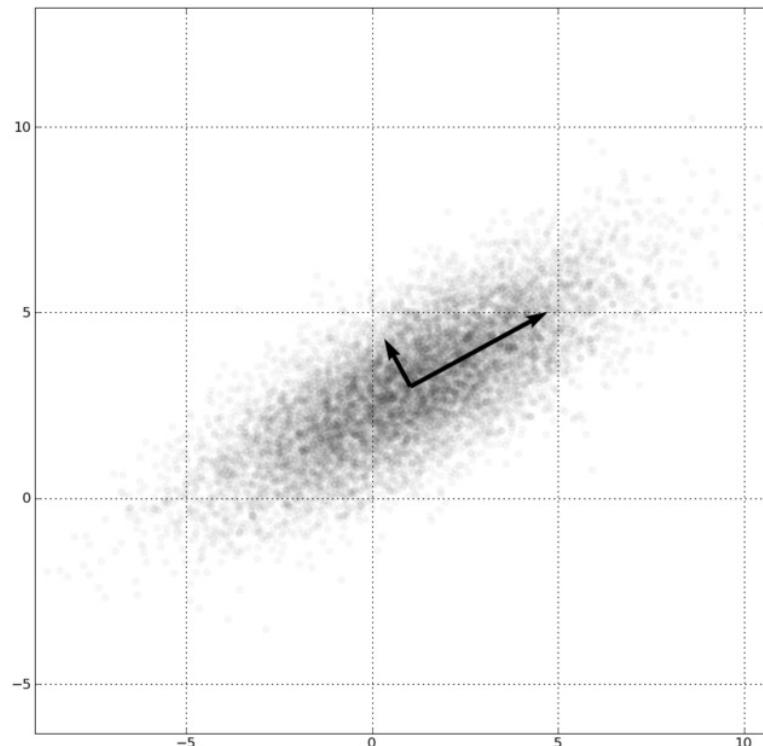
# Principal Component Analysis

**Идея 1:** давайте выделять в пространстве признаков направления, вдоль которых разброс точек наибольший (они кажутся наиболее информативными)

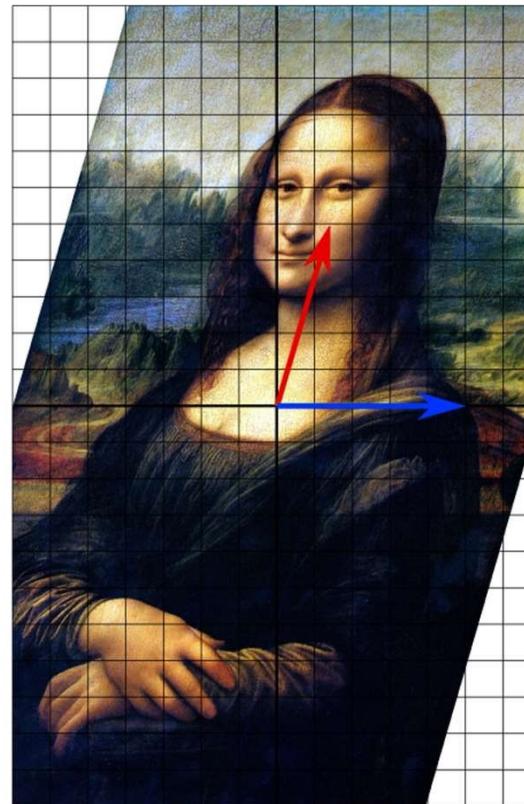
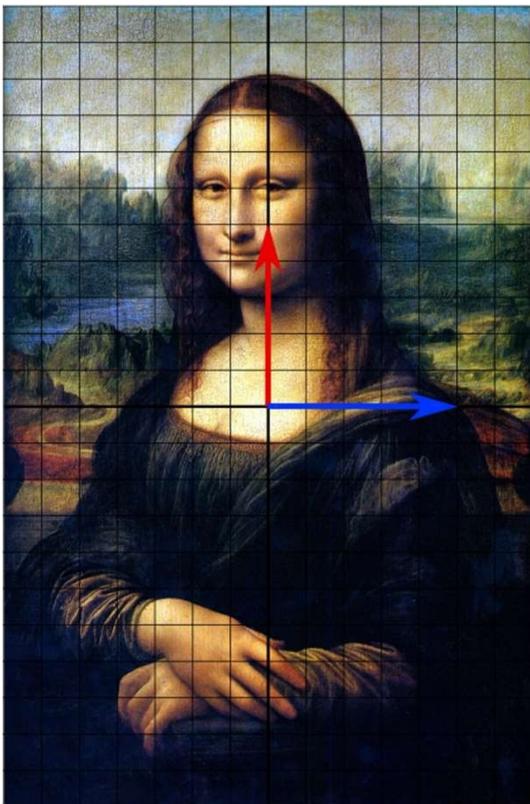


# Principal Component Analysis

**Идея 3.** Переход в базис, в котором матрица ковариаций диагональна



# Собственные векторы



# Собственные векторы

- $A$  — матрица размера  $n \times n$
- Пусть  $Ax = \lambda x$
- Тогда  $x$  — собственный вектор,  $\lambda$  — собственное значение
- $x$  — вектор, который не меняет направление под действием матрицы

# Решение

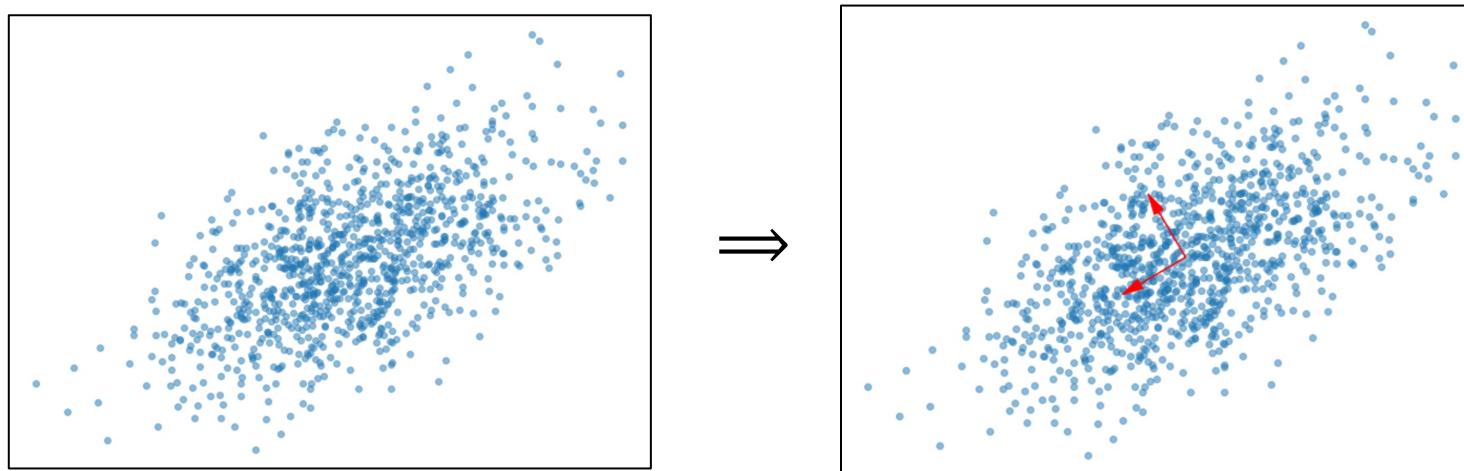
- Столбцы  $W$  — собственные векторы матрицы  $X^T X$ , соответствующие наибольшим собственным значениям  $\lambda_1, \lambda_2, \dots, \lambda_d$
- $\frac{\sum_{i=1}^d \lambda_i}{\sum_{i=1}^D \lambda_i}$  — доля дисперсии, сохранённой при понижении размерности

# Возврат в исходное пространство

$$Z = XW^T$$

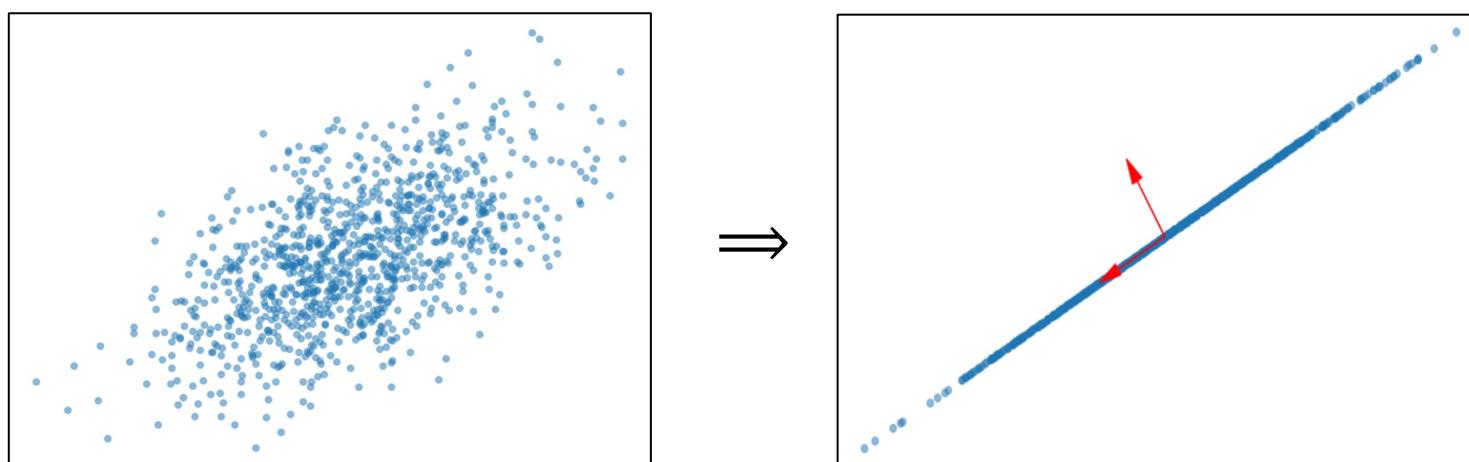
$W^T$  – матрица перехода к новому базису (размерность пространства не уменьшаем). Т.к. базис ортонормированный, то  $W^T$  – ортогональная матрица, т.е.

$$W^T W = I \Rightarrow X = \underbrace{XW^T W}_Z$$



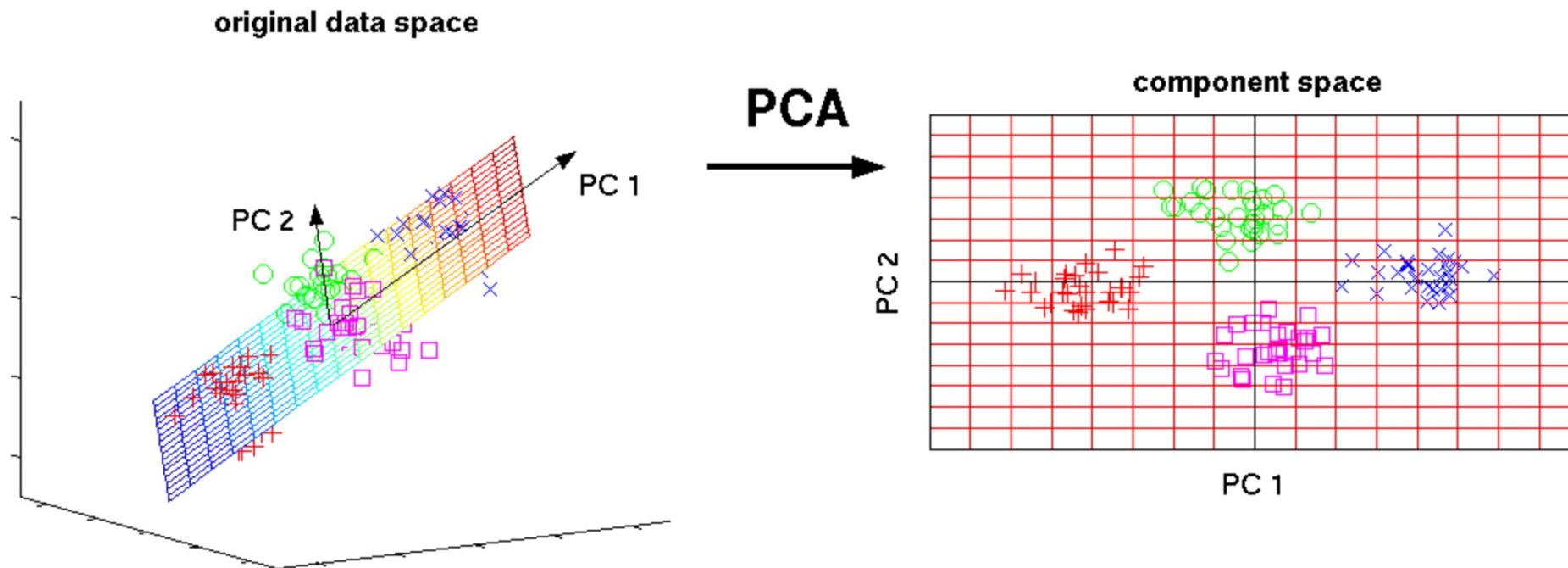
# Возврат в исходное пространство

$$\hat{X}_{n \times D} = X_{n \times D} W^T_{D \times D} W_{D \times D}$$



# PCA (интерпретация 2)

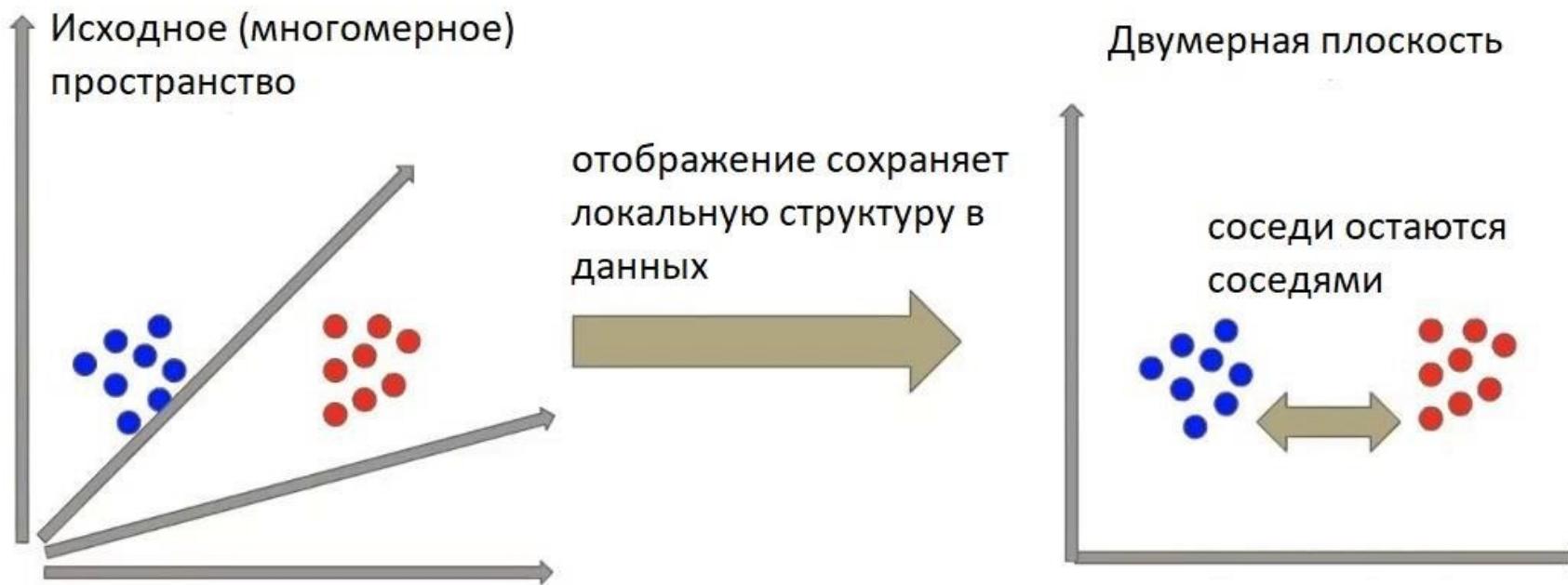
**Идея 2:** давайте строить проекцию выборки на линейное подпространство меньшей размерности. А выбирать его так, чтобы квадраты отклонений точек от проекций были минимальны.



# t-SNE

*t-SNE – t-distributed stochastic neighbor embedding*

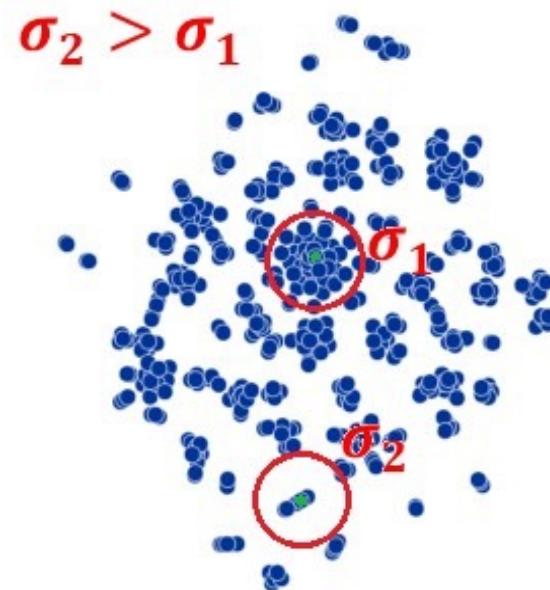
Метод нелинейного снижения размерности пространства признаков



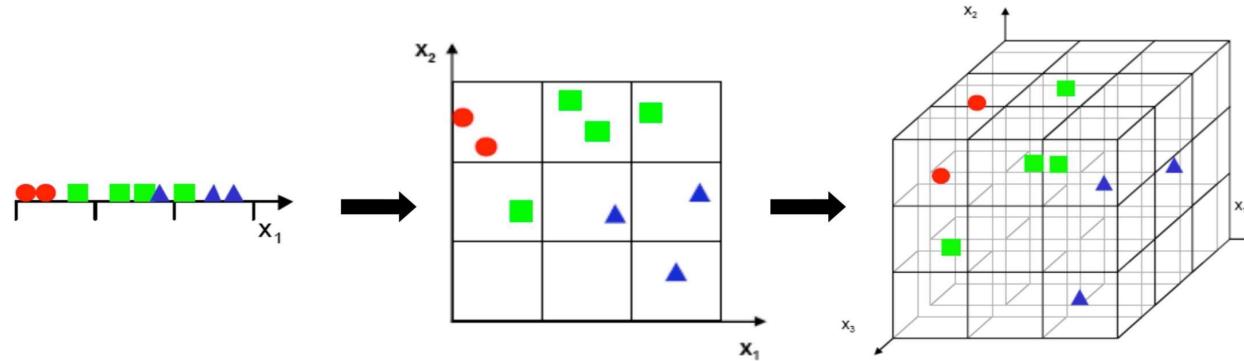
## БЛИЗОСТЬ ОБЪЕКТОВ В ИСХОДНОМ ПРОСТРАНСТВЕ

$$p(i|j) = \frac{\exp(-\|x_i - x_j\|^2 / 2\sigma_j^2)}{\sum_{k \neq j} \exp(-\|x_k - x_j\|^2 / 2\sigma_j^2)}$$

- объекты из окрестности  $x_j$  приближаются нормальным распределением
- чем кучнее объекты из этой окрестности, тем меньше берётся значение  $\sigma_j^2$



## БЛИЗОСТЬ ОБЪЕКТОВ В НОВОМ ПРОСТРАНСТВЕ

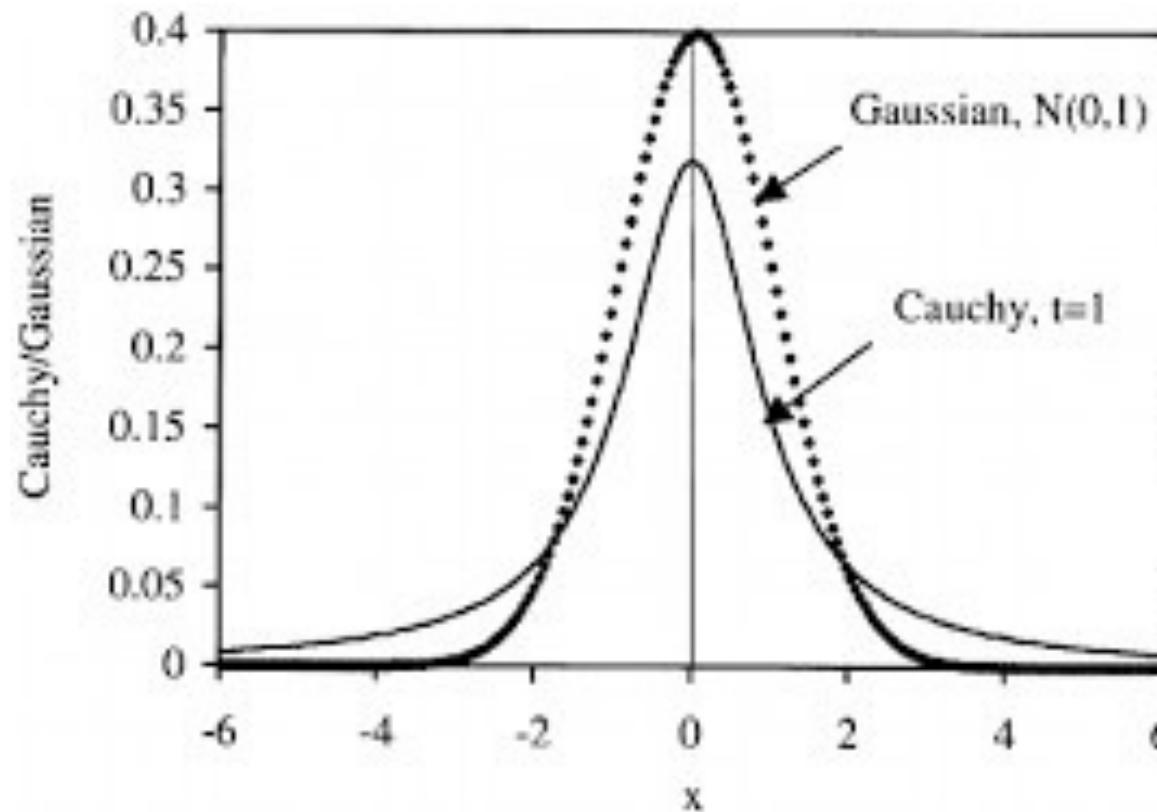


- Будем измерять сходство объектов в новом пространстве с помощью распределения Коши, так как оно не так сильно штрафует за увеличение расстояний между объектами:

$$q(i|j) = \frac{\left(1 + \|z_i - z_j\|^2\right)^{-1}}{\sum_{k \neq j} \left(1 + \|z_k - z_j\|^2\right)^{-1}}$$

## НОРМАЛЬНОЕ РАСПРЕДЕЛЕНИЕ И РАСПРЕДЕЛЕНИЕ КОШИ

- Будем измерять сходство объектов в новом пространстве с помощью распределения Коши, так как оно не так сильно штрафует за увеличение расстояний между объектами:

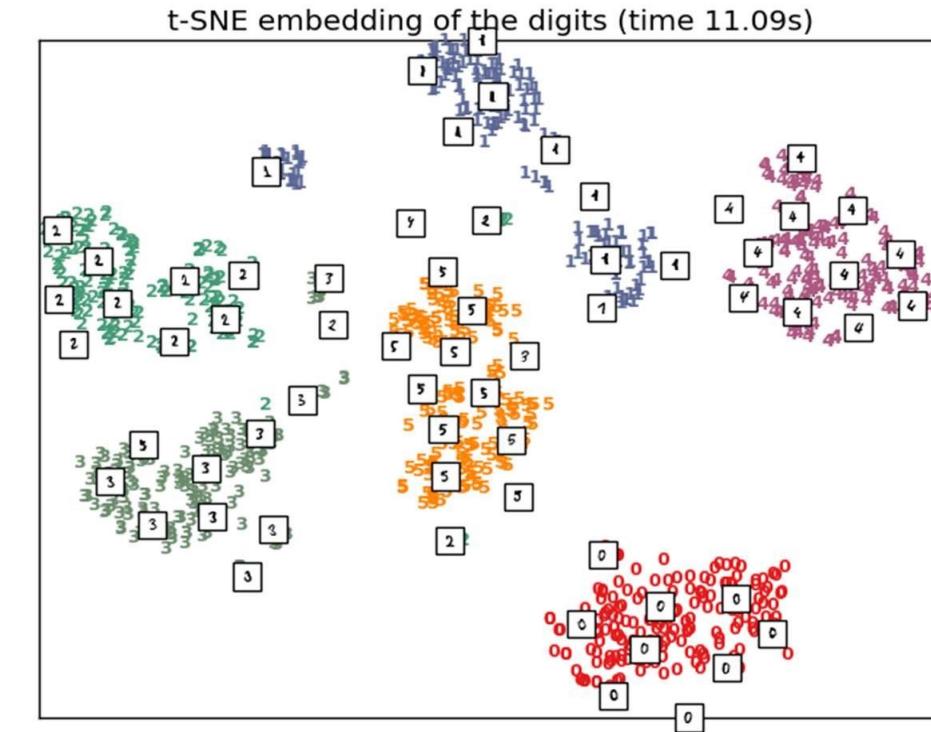
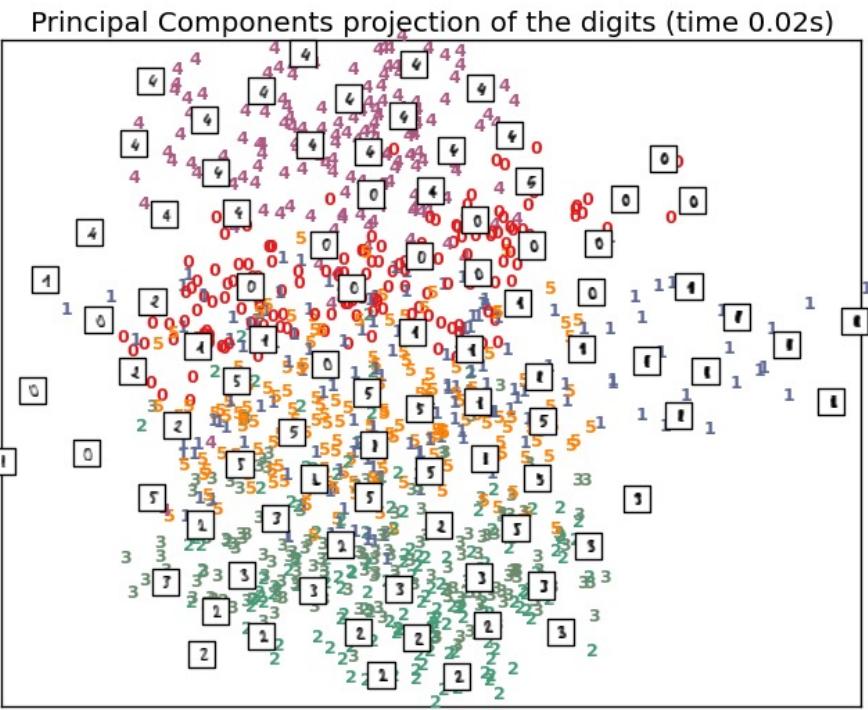


## Обучение t-SNE

- Для построения проекций  $z_i$  объектов  $x_i$  будем минимизировать расстояние между исходным и полученным распределениями (минимизируем дивергенцию Кульбака-Лейблера).

$$KL(p||q) = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}} \rightarrow \min_{z_1, \dots, z_n}$$

# PCA и t-SNE



# Недостатки t-SNE

- Вычислительно сложный
- При добавлении новых точек нужно переобучать алгоритм
- Недетерминированный – от запуска к запуску получаем разные результаты

# Кластеризация

# Обучение с учителем (supervised learning)

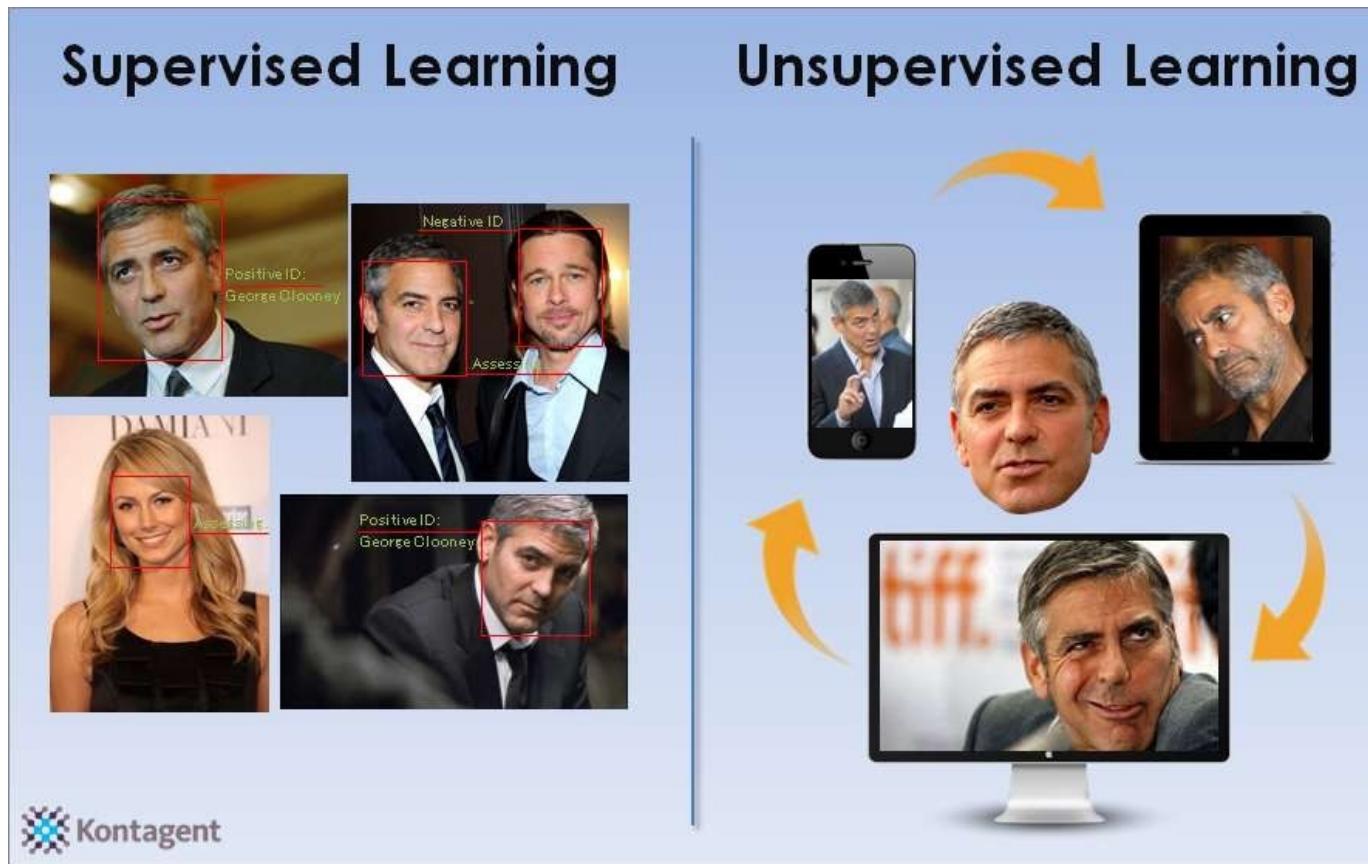
- Для каждого объекта известен ответ (класс или число)
- Даны примеры объектов с ответами
- Нужно построить модель, которая будет предсказывать ответы для новых объектов

# Обучение без учителя (unsupervised learning)

- Даны объекты
- Нужно найти в них внутреннюю структуру
- Примеры:
  - Кластеризация
  - Обнаружение аномалий
  - Тематическое моделирование
  - Визуализация
  - ...
- Ближе к обучению в реальной жизни

Тематическое моделирование — это метод анализа текстовых данных, который используется для выявления скрытых тем в большом объеме текстовой информации.

# Обучение с учителем и без учителя



# Кластеризация

- Дано: матрица «объекты-признаки»  $X$
- Найти:
  1. Множество кластеров  $Y$
  2. Алгоритм кластеризации  $a(x)$ , который приписывает каждый объект к одному из кластеров
- Каждый кластер состоит из похожих объектов
- Объекты из разных кластеров существенно отличаются

# Отличия

## Обучение с учителем

- Цель: минимизация функционала ошибки
- Множество ответов известно заранее
- Конкретные способы измерения качества

## Кластеризация

- Нет строгой постановки
- Множество кластеров неизвестно
- Правильные ответы отсутствуют— сложно измерить качество

# Зачем кластеризовать?

- Маркетинг: выявление типичных групп покупателей
- Модерация: проверка только одного сообщения из кластера
- UI/UX: каталогизация фотографий (по людям, по местам, ...)

# Виды кластеризации

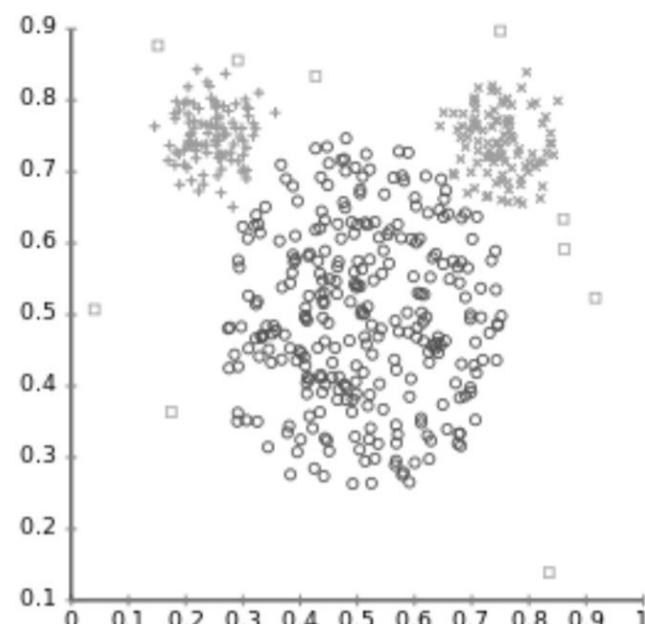
# Форма кластеров



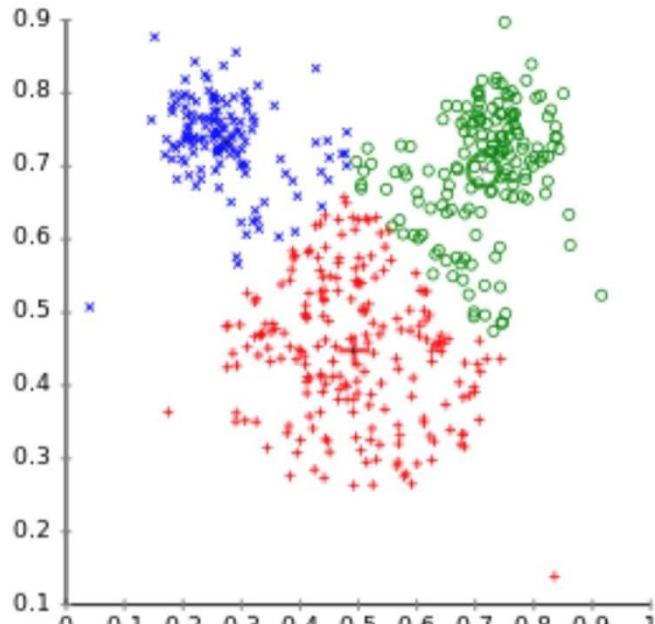
# Форма кластеров



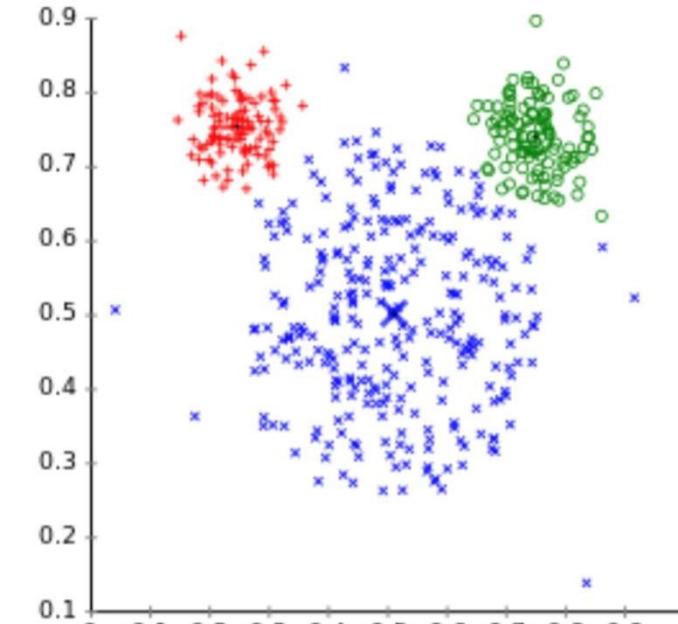
# Различия в результатах работы



Исходная выборка  
("Mouse" dataset)



Метод 1



Метод 2

Text  
Text  
Text

# «Жесткая» и «мягкая» кластеризации

# Кластеризация для выделения «тем»

A 3D perspective word cloud centered on the words "Mathematical finance". The words are rendered in various colors (red, blue, green, yellow) and sizes, creating a sense of depth as they recede into the background.

The word cloud illustrates the following concepts:

- CLUSTERING**: network, significantly, isolated, vertices, triangle, induced, open, black, weighted, proposed, formula, preceding, part, attempt, tends, randomly, directed, generalisation, missing.
- COEFFICIENT**: network, significantly, isolated, vertices, triangle, induced, open, black, weighted, proposed, formula, preceding, part, attempt, tends, randomly, directed, generalisation, missing.
- Other terms**: FORMALLY, VERSION, SEXIST, CLIQUE, SET, PROPORTION, DEFINITIONS, TRIPLET, PATH, COMPUTED, SEGMENTS, NODE, CENTRED, INCIDENT, LIKELIHOOD, THEORY, THICK, INDICATION, DENSITY, RANDOM, BINARY, IDENTICAL, INTRODUCED, MODULAR, TIED, NUMBER, CONSISTS, CONSTRUCTED, DIVIDED, CHARACTERISED, AVERAGE, NEIGHBOUR, CLUSTER, REALIZED, QUALITIES, LOCAL, CONNECTION, NEIGHBOURHOOD, PROBABILITY.

В жесткой кластеризации каждый объект принадлежит только одному кластеру. Это означает, что объекты четко разделены на группы, и нет перекрытия между кластерами. Примеры методов жесткой кластеризации включают:

**K-средних (K-means):** алгоритм, который делит данные на K кластеров, минимизируя внутрикластерные расстояния.

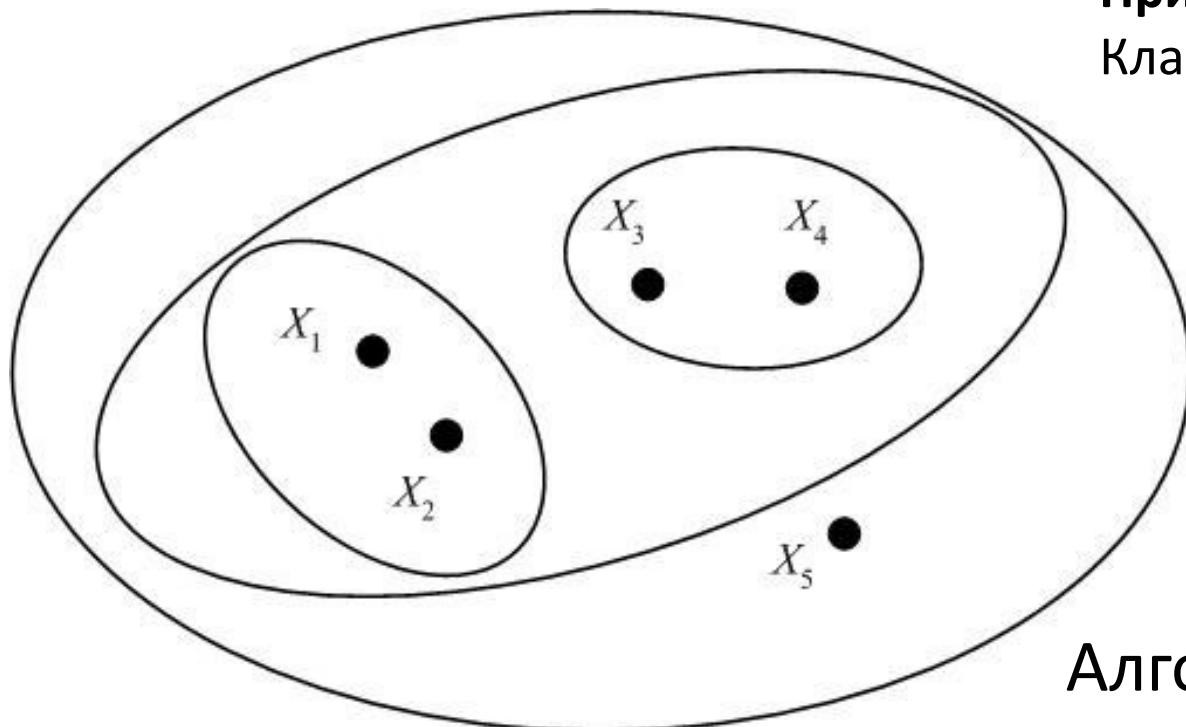
Иерархическая кластеризация: создает дерево кластеров, где на каждом уровне можно видеть, как объекты группируются.

В мягкой кластеризации объекты могут принадлежать нескольким кластерам с определенной вероятностью или степенью принадлежности. Это позволяет более гибко учитывать неопределенности и перекрытия между кластерами. Примеры методов мягкой кластеризации включают:

Fuzzy C-means: аналог К-средних, но каждый объект имеет степень принадлежности к каждому кластеру.

Gaussian Mixture Models (GMM): предполагает, что данные распределены по нескольким гауссовым распределениям, и каждый объект имеет вероятность принадлежности к каждому из них.

# Иерархическая кластеризация



Пример:

Кластеризация статей на Хабре

IT

Алгоритмы

Алгоритмы  
и структуры  
данных

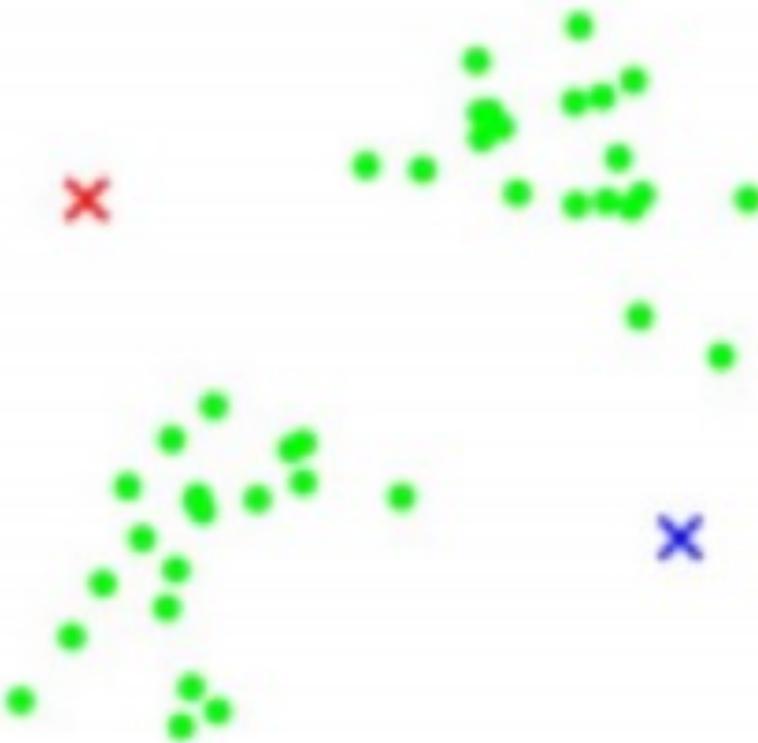
Методы  
машинного  
обучения

# K-Means

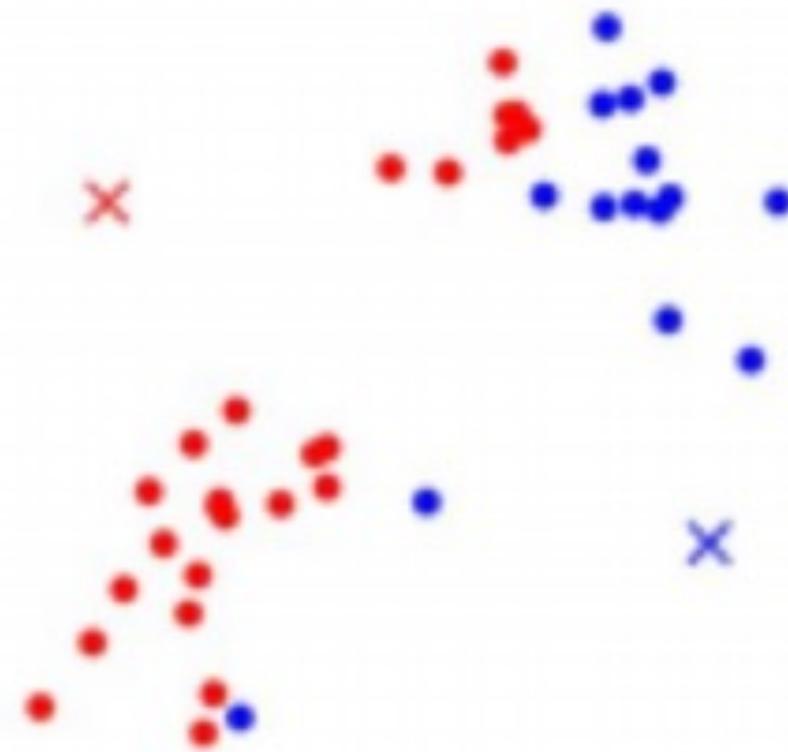
# K Means



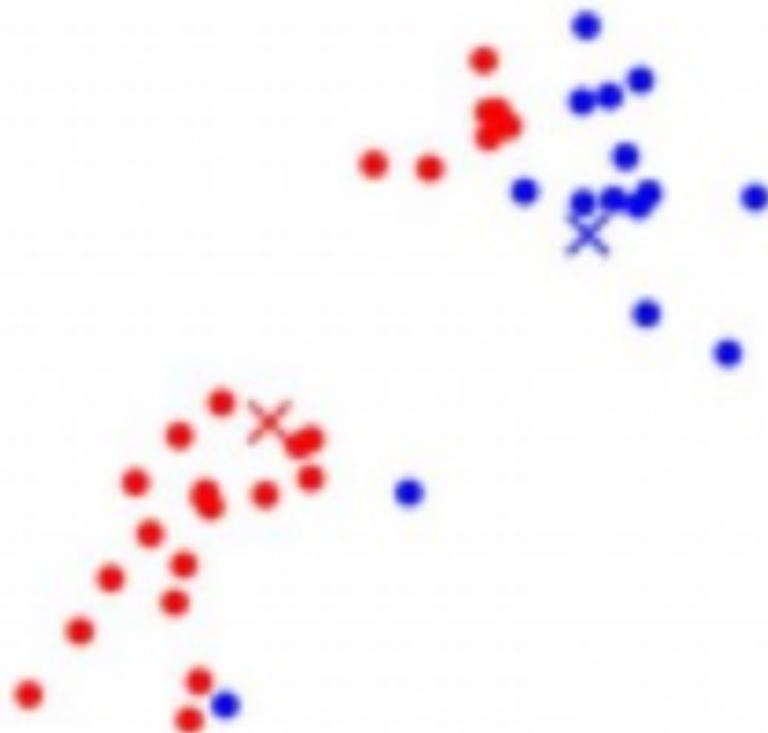
# Как работает K Means



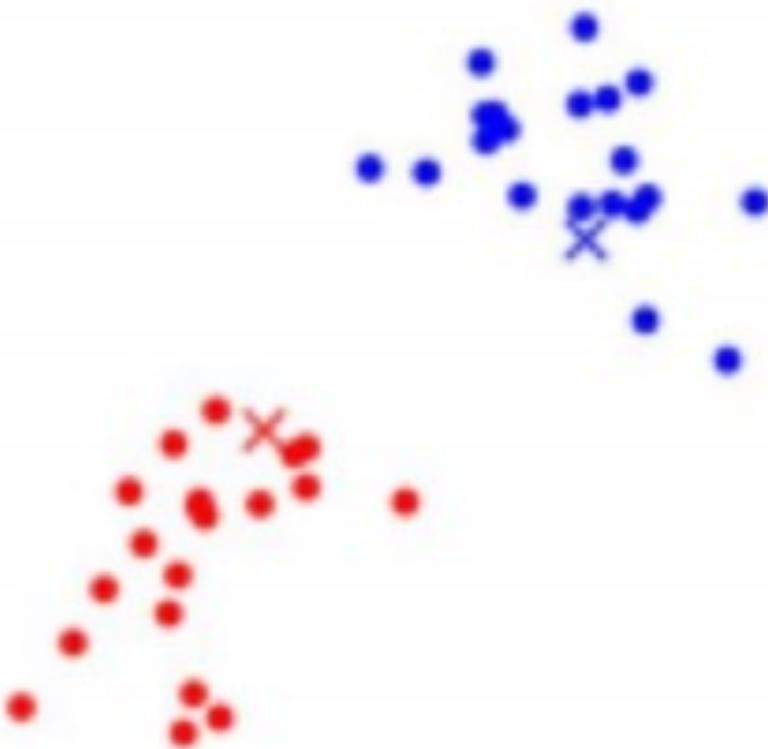
# Как работает K Means



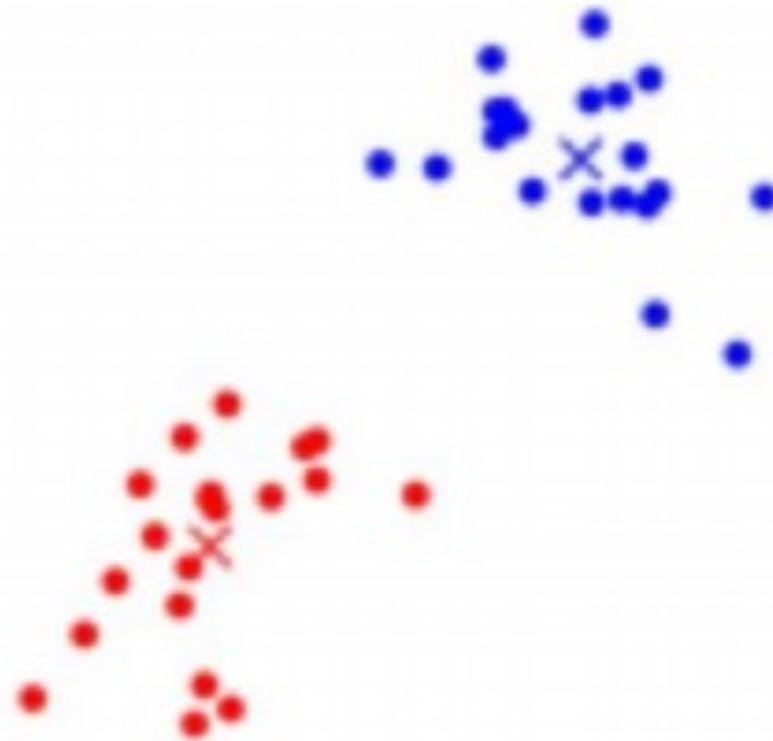
# Как работает K Means



# Как работает K Means



# Как работает K Means



# K-Means

- Дано: выборка  $x_1, \dots, x_\ell$
- Параметр: число кластеров  $K$
- Начало: случайно выбрать  $K$  центров кластеров  $c_1, \dots, c_K$
- Повторять по очереди до сходимости:

- Шаг А: отнести каждый объект к ближайшему центру

$$y_i = \arg \min_{j=1, \dots, K} \rho(x_i, c_j)$$

- Шаг Б: переместить центр каждого кластера в центр тяжести

$$c_j = \frac{\sum_{i=1}^{\ell} x_i [y_i = j]}{\sum_{i=1}^{\ell} [y_i = j]}$$

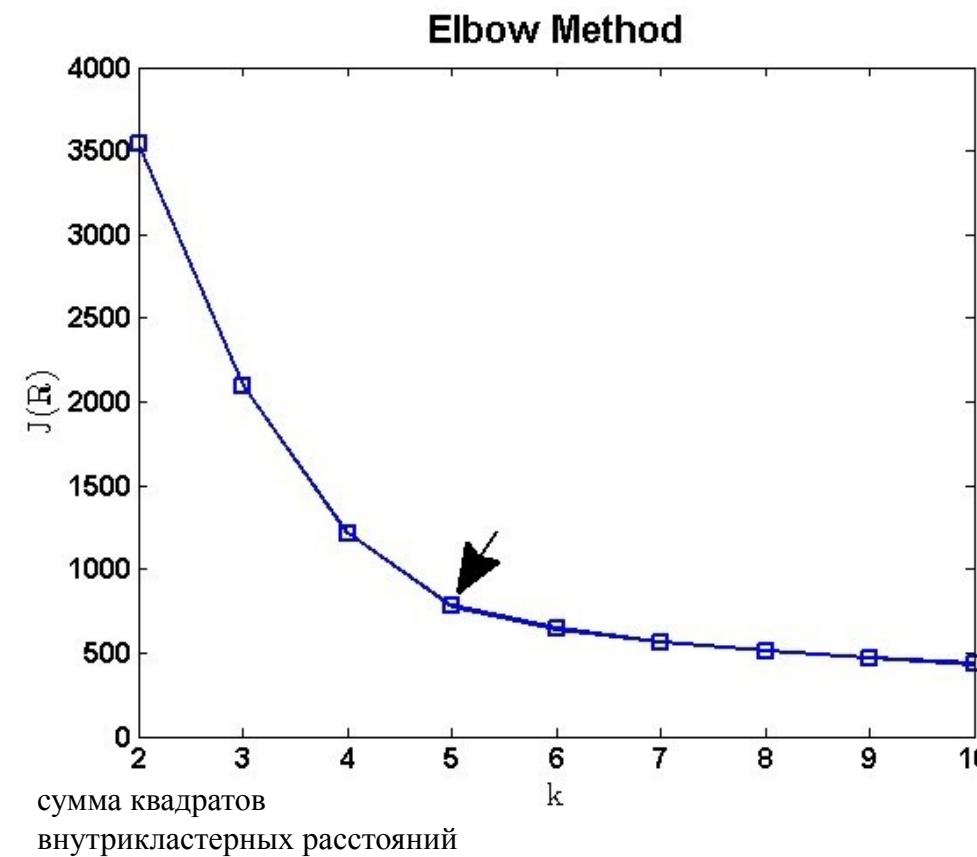
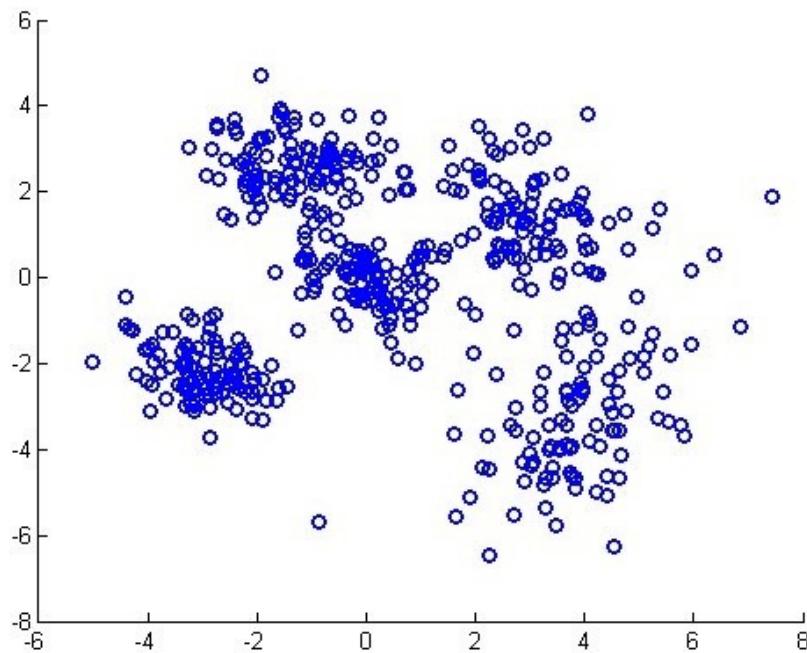
# Выбор числа кластеров

- Качество кластеризации: внутрикластерное расстояние

$$J(C) = \sum_{i=1}^{\ell} \rho(x_i, c_{y_i})$$

- Зависит от  $K$
- Нужно подобрать такое  $K$ , после которого качество меняется не слишком сильно

# Выбор числа кластеров



# Особенности K-Means

- Может работать с большими объёмами данных
- Подходит для кластеров с простой геометрией
- Требует выбора числа кластеров

Силуэт как метрика качества

Для каждого объекта  $i$  рассчитываются два значения:

a( $i$ ): Среднее расстояние от объекта  $i$  до всех других объектов в том же кластере. Это значение показывает, насколько близки объекты внутри одного кластера.

b( $i$ ): Среднее расстояние от объекта  $i$  до всех объектов в ближайшем кластере (кластере, отличном от того, к которому принадлежит объект  $i$ ). Это значение показывает, насколько близки объекты к другим кластерам.

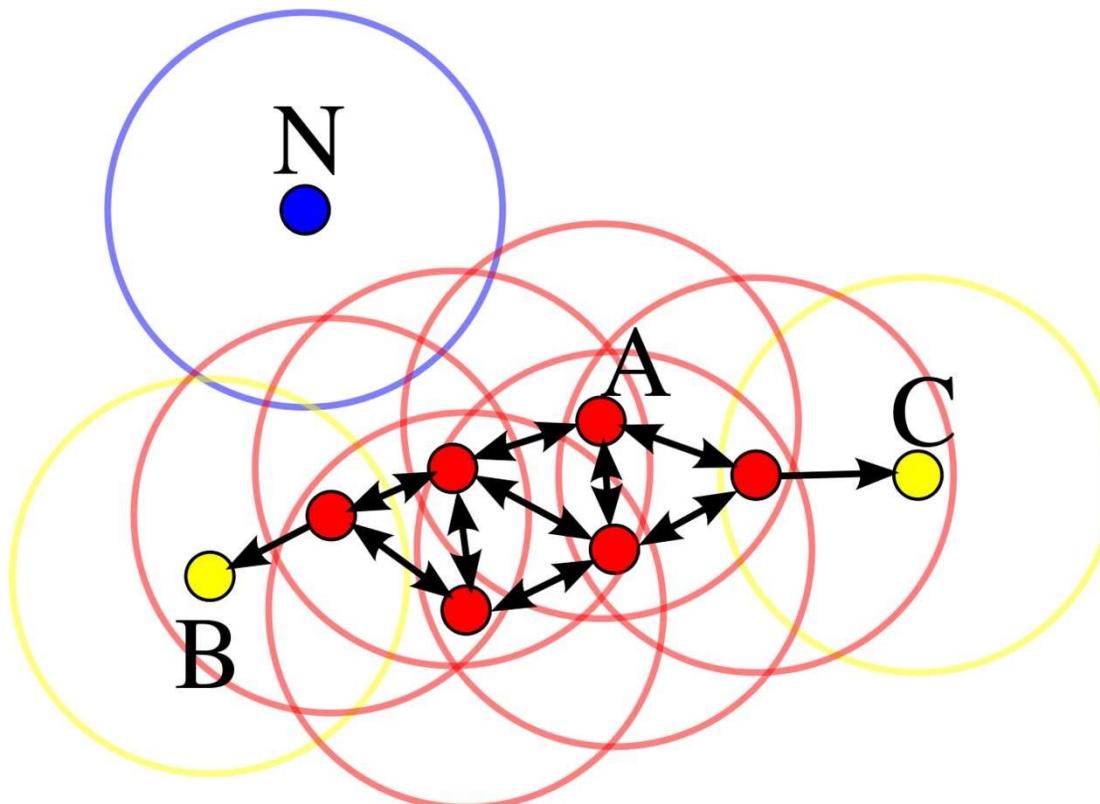
Силуэт для объекта  $i$  рассчитывается по формуле:

$$(b - a) / \max(a, b)$$

# Density-based clustering

Density-based clustering (кластеризация на основе плотности) — это метод кластеризации, который группирует данные на основе плотности распределения точек в пространстве.

# Основные, граничные и шумовые точки



**Основные (ядровые) точки:** Это точки, которые имеют не менее заданного числа соседей (MinPts) в пределах радиуса  $\epsilon$  (epsilon). Основные точки служат центрами кластеров. Они могут инициировать формирование кластера, так как вокруг них достаточно плотности точек.

**Граничные точки:** Это точки, которые находятся в пределах радиуса  $\epsilon$  от основной точки, но не имеют достаточного количества соседей, чтобы быть основной точкой (т.е. у них меньше MinPts соседей).

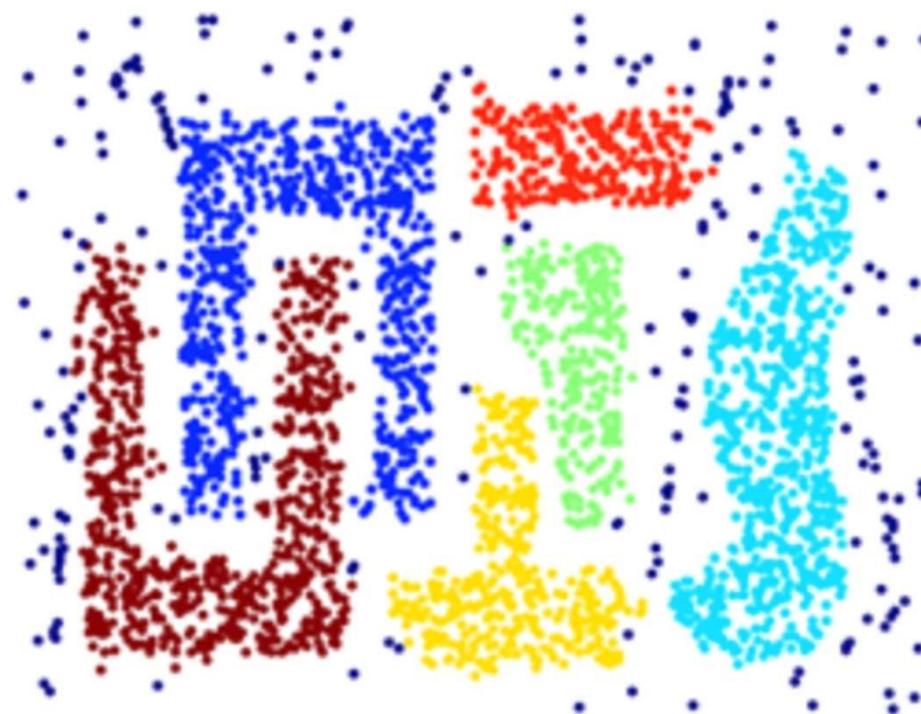
**Шумовые точки:** Это точки, которые не являются ни основными, ни граничными. Они не имеют достаточной плотности соседей, чтобы быть частью кластера.

# Параметры DBSCAN

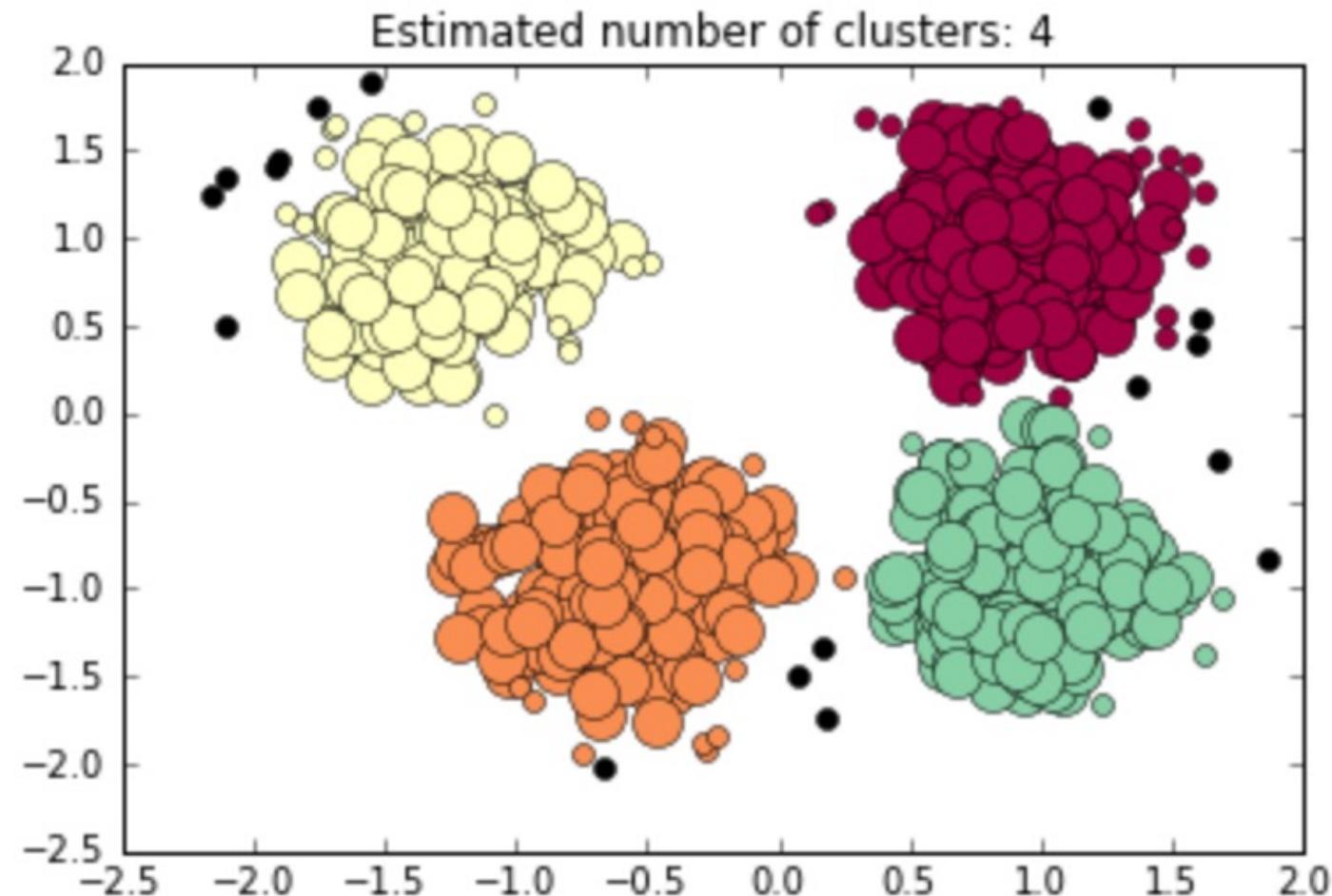
DBSCAN (Density-Based Spatial Clustering of Applications with Noise) — это алгоритм кластеризации, основанный на плотности, который используется для выявления кластеров в данных с произвольной формой и для обработки выбросов.

- Размер окрестности (eps)
- Минимальное число объектов в окрестности — для определения основных точек

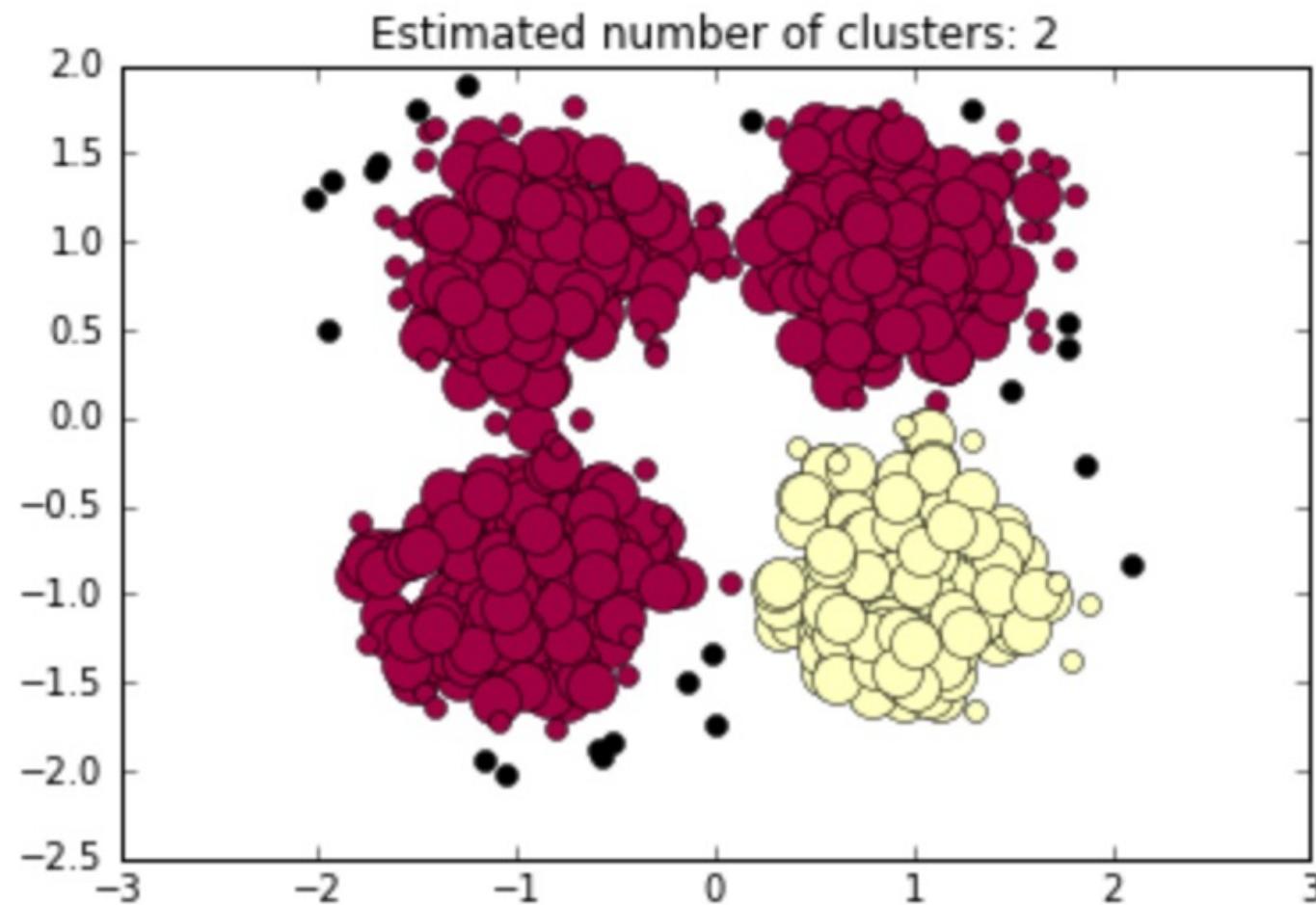
# DBSCAN: результаты работы



# Определение числа кластеров в DBSCAN



# Определение числа кластеров в DBSCAN



# Особенности DBSCAN

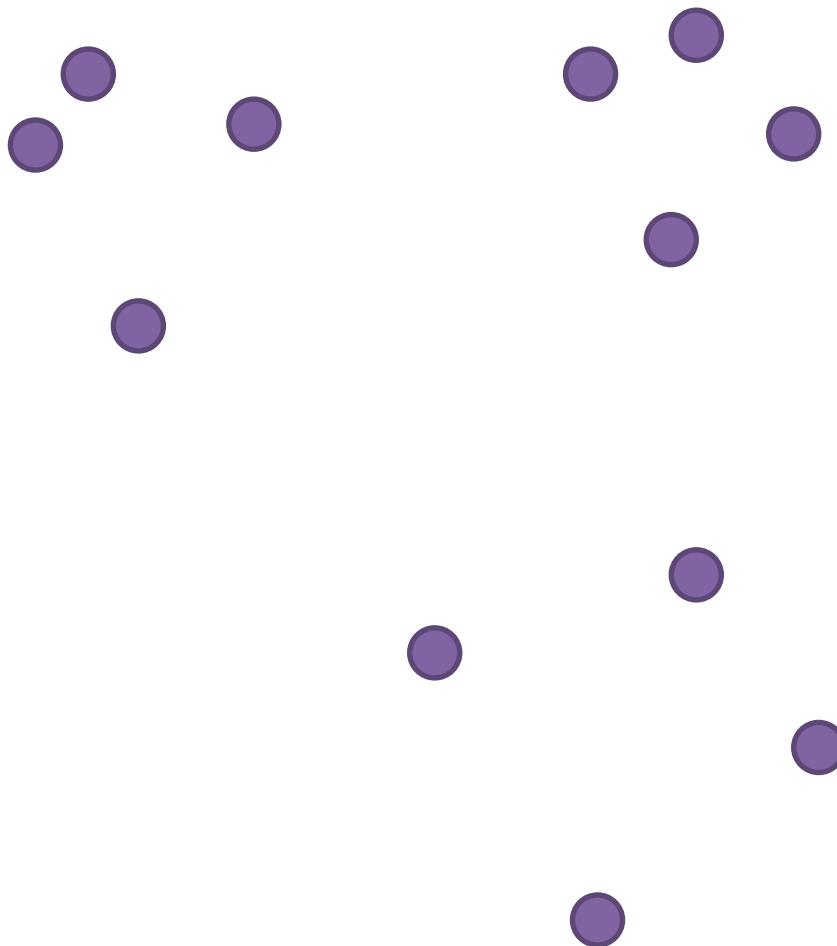
- Находит кластеры произвольной формы
- Может работать с большими объёмами данных
- Нужно подбирать размер окрестности ( $\text{eps}$ ) и минимальное число объектов в окрестности

# Иерархическая кластеризация

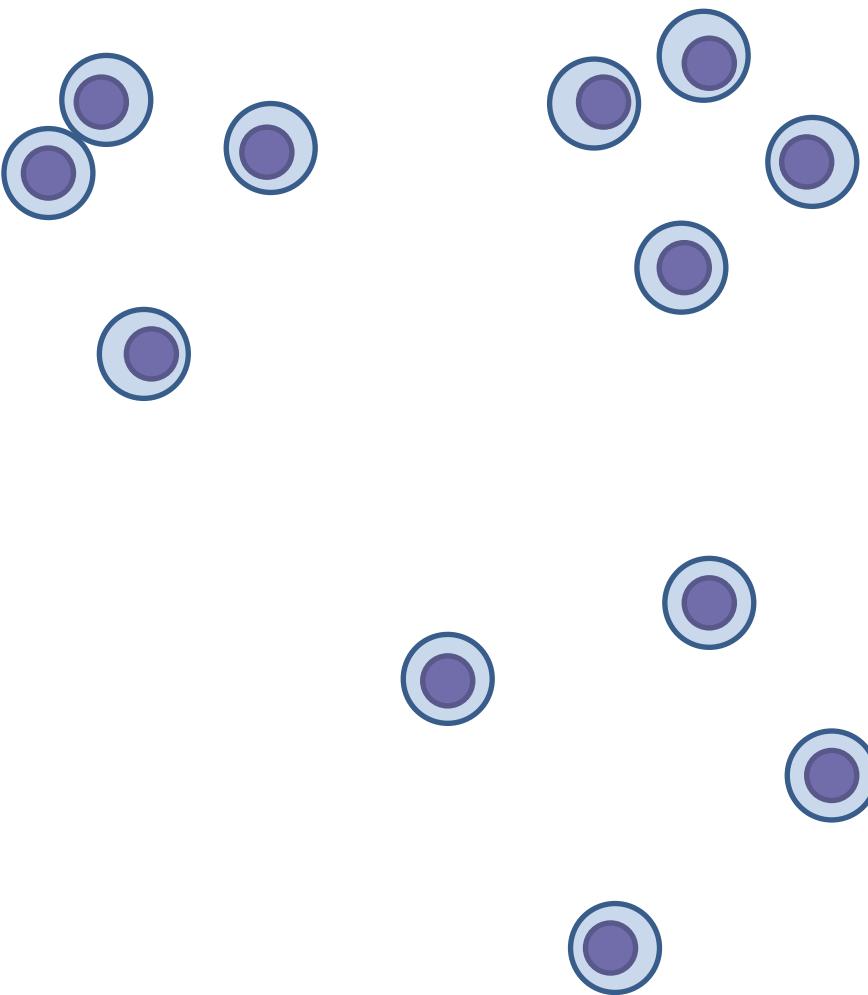
# Виды иерархической кластеризации

- Аггломеративная – на каждой итерации объединяем два меньших кластера в один побольше
- Дивизивная – на каждой итерации делим один большой кластер на два поменьше

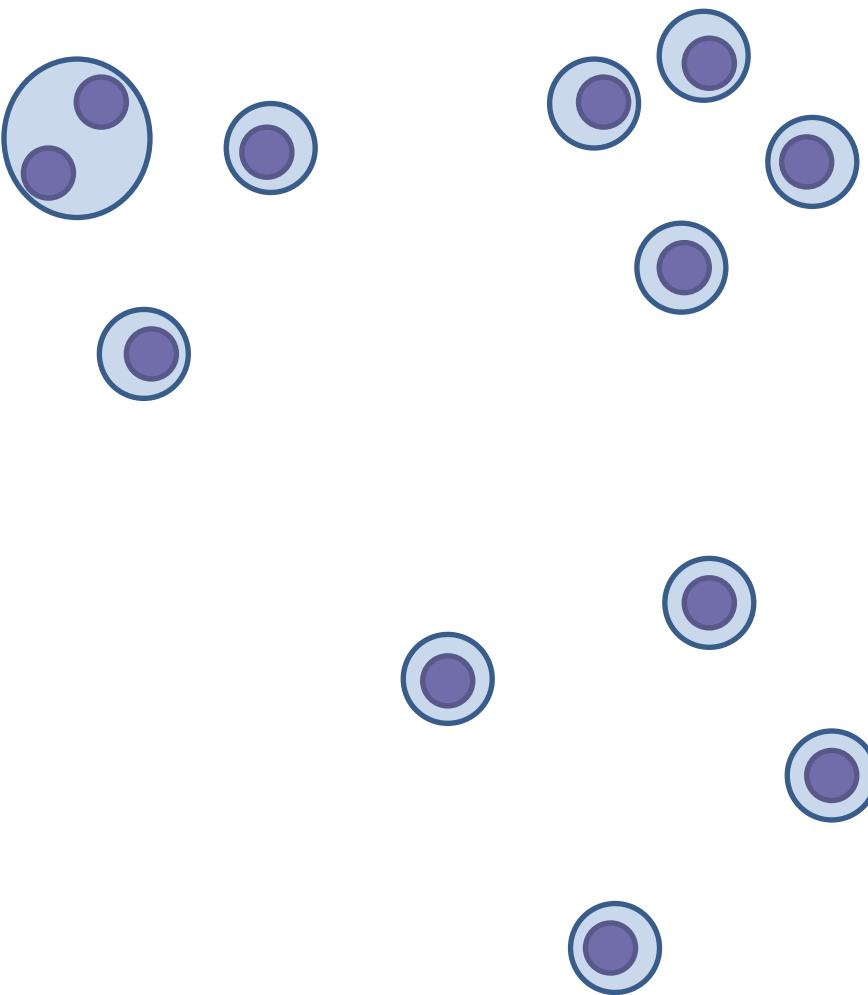
# Агломеративная кластеризация



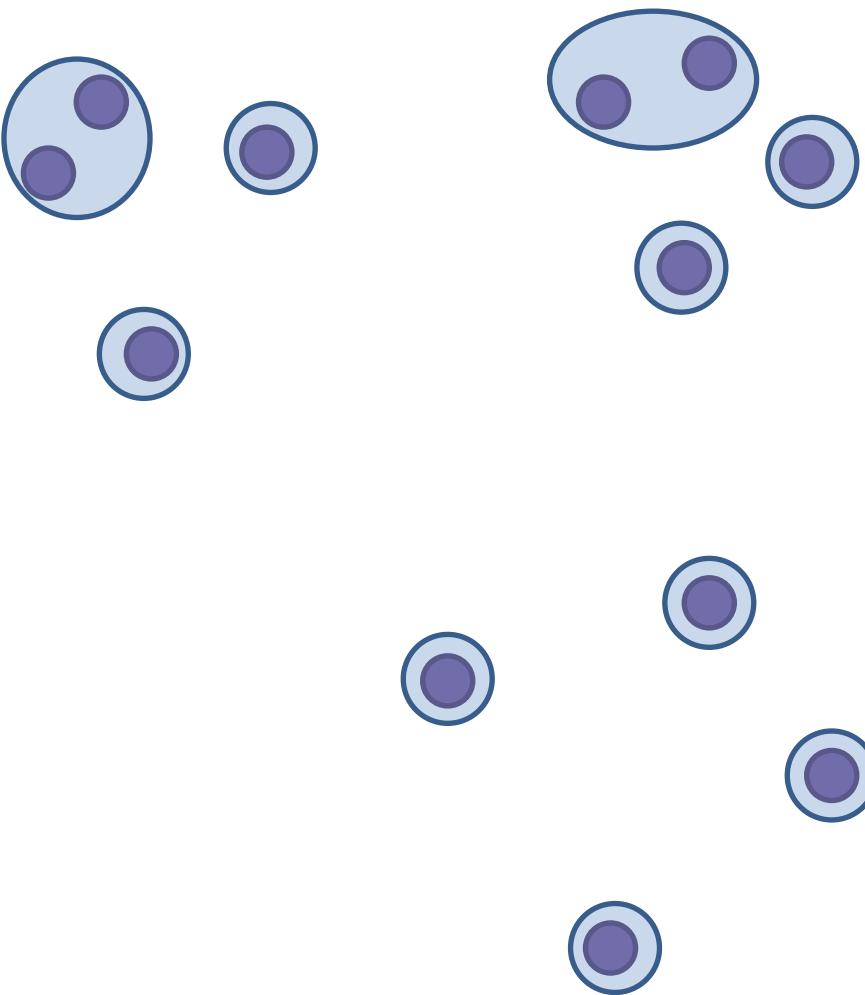
# Агломеративная кластеризация



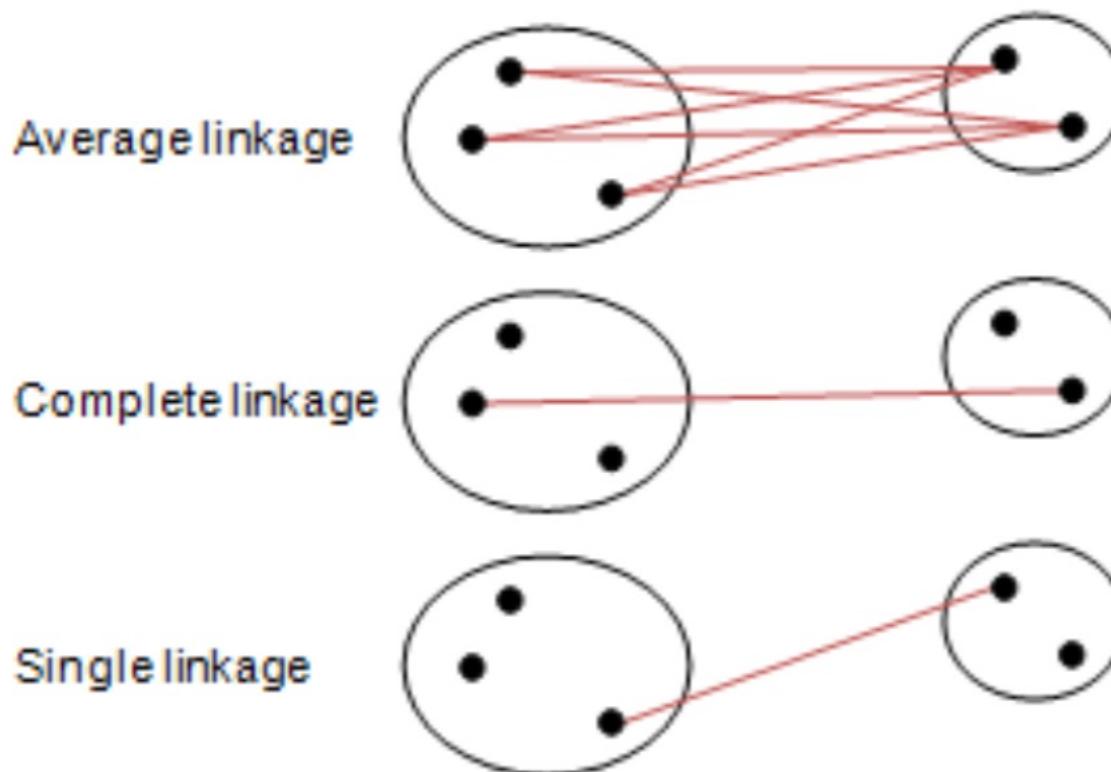
# Агломеративная кластеризация



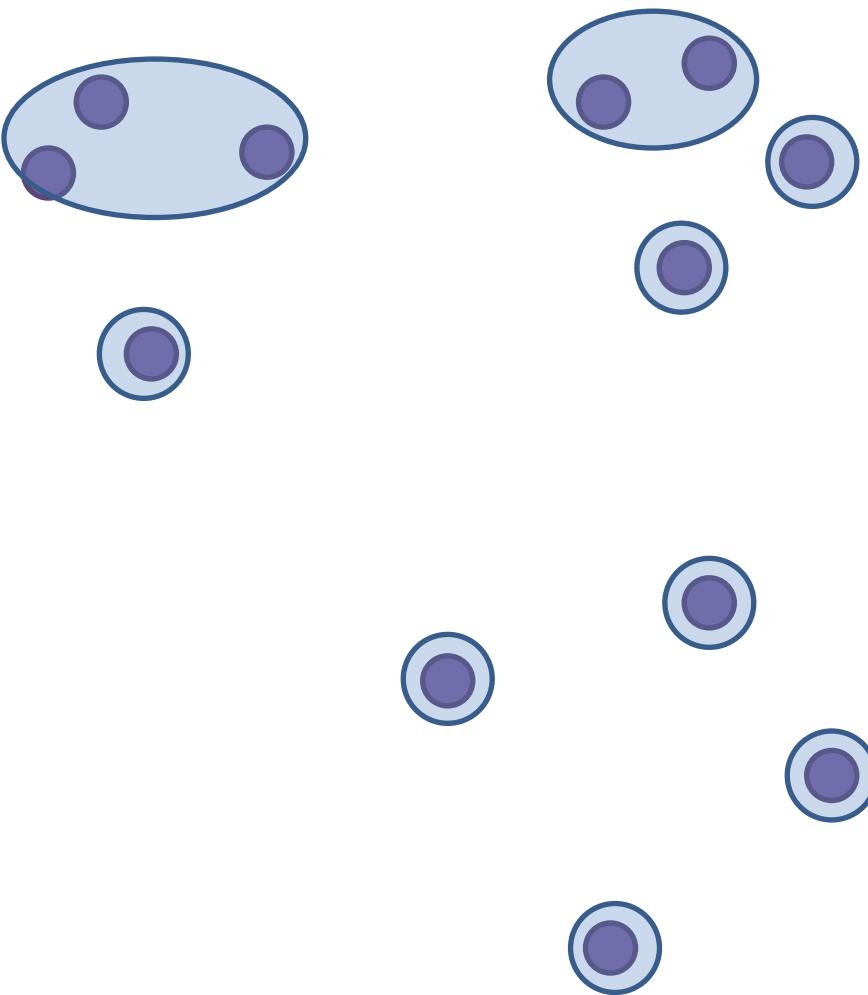
# Агломеративная кластеризация



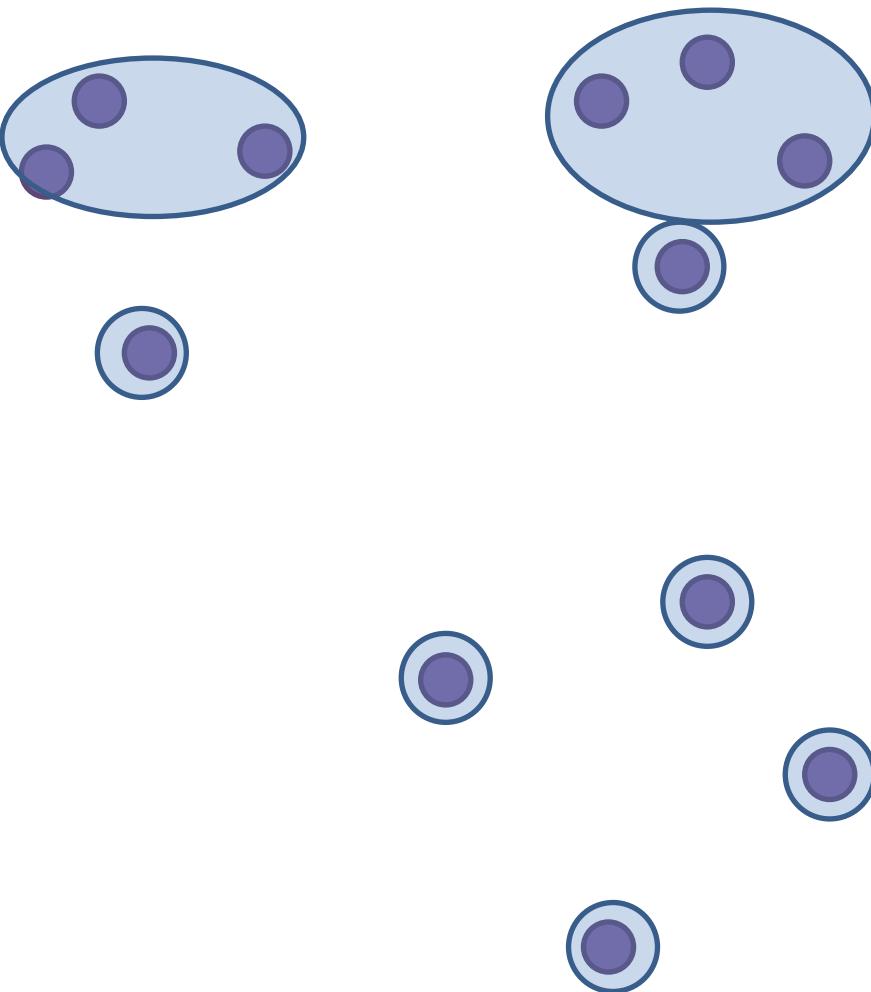
# Расстояния между кластерами



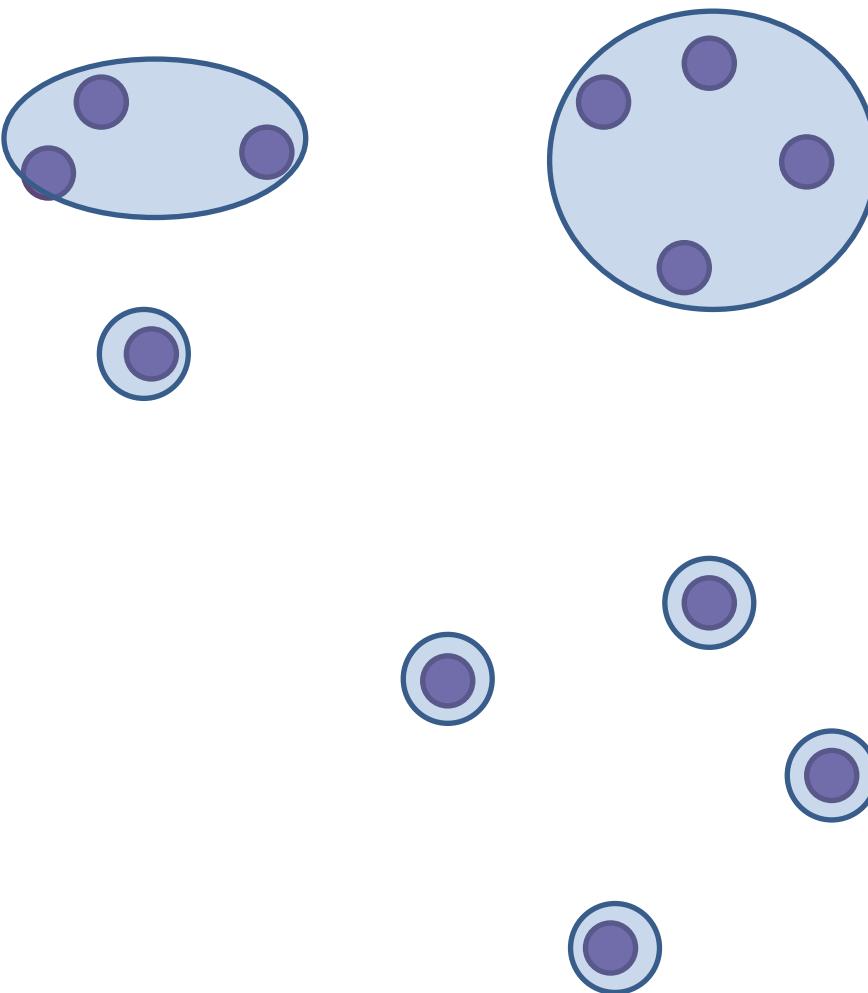
# Агломеративная кластеризация



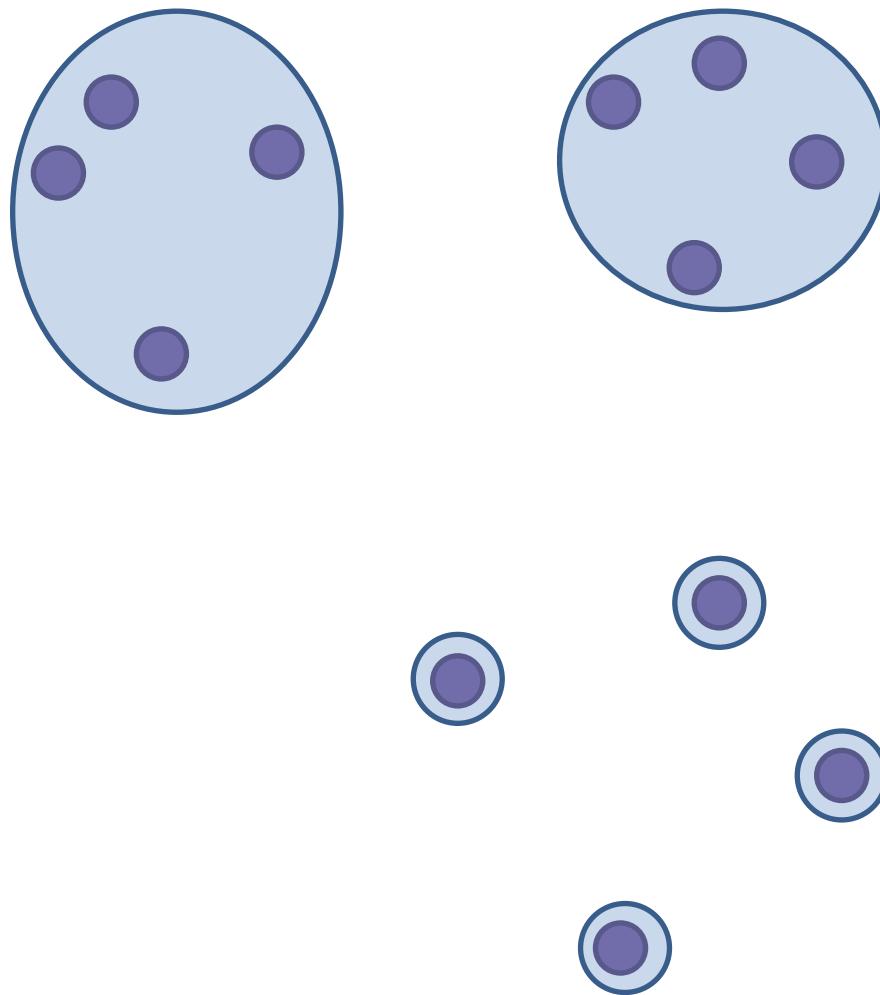
# Агломеративная кластеризация



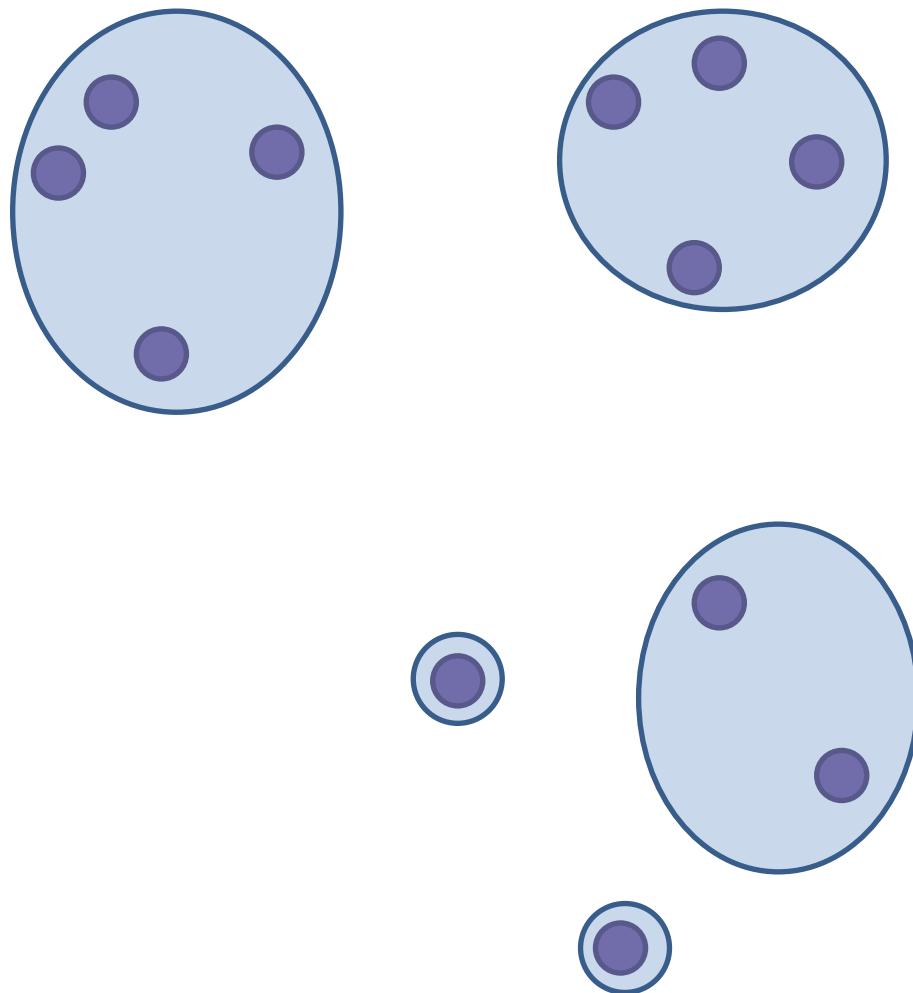
# Агломеративная кластеризация



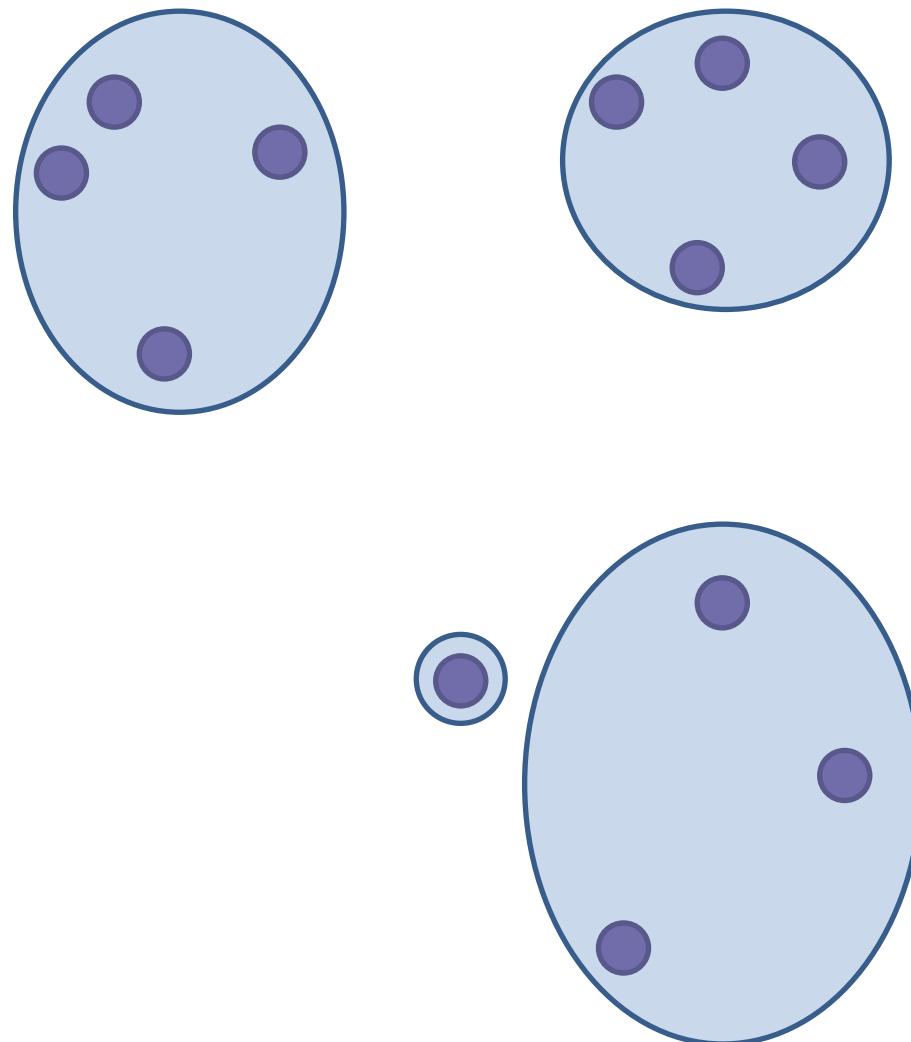
# Агломеративная кластеризация



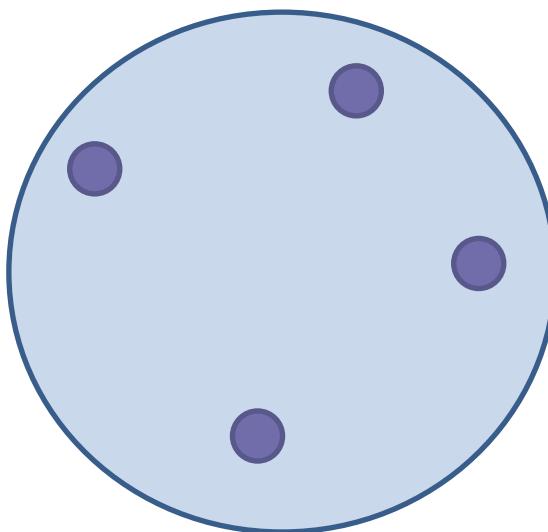
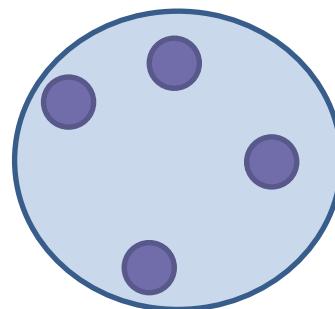
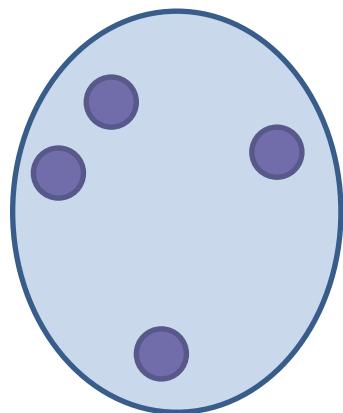
# Агломеративная кластеризация



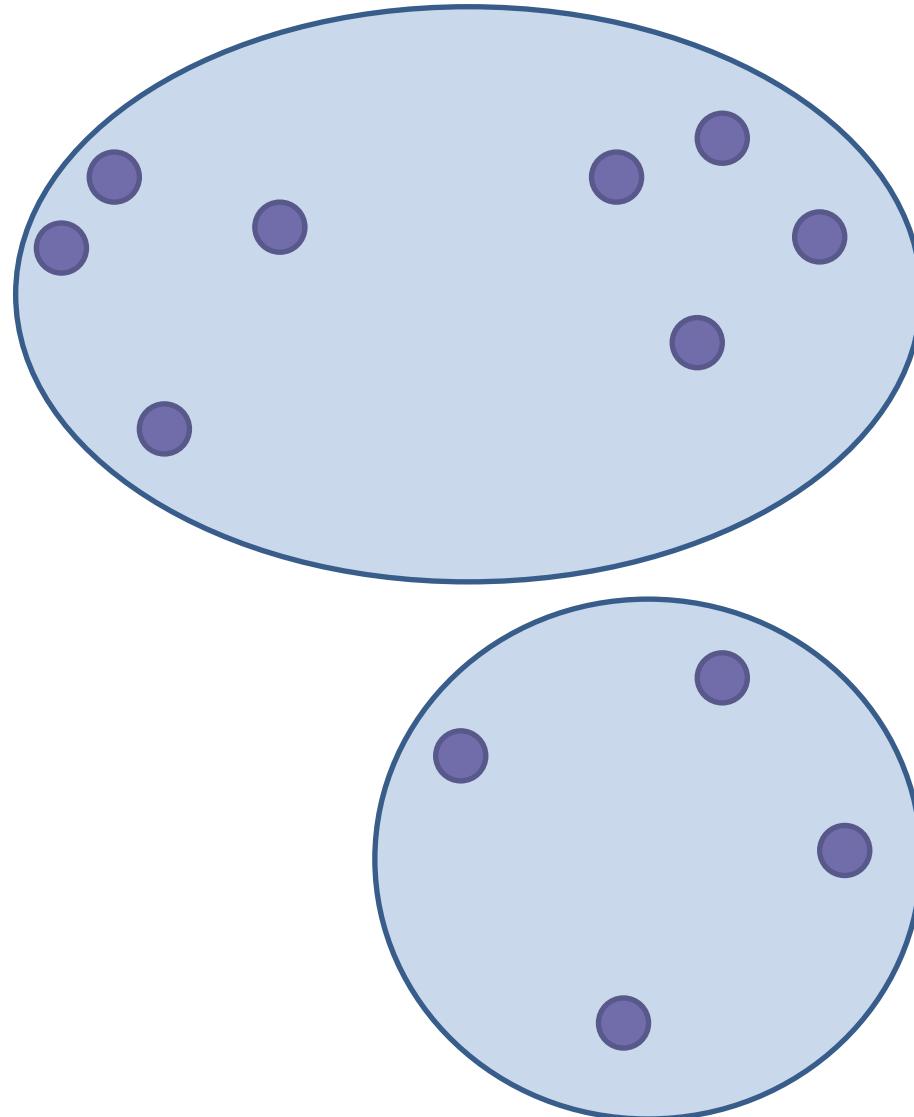
# Агломеративная кластеризация



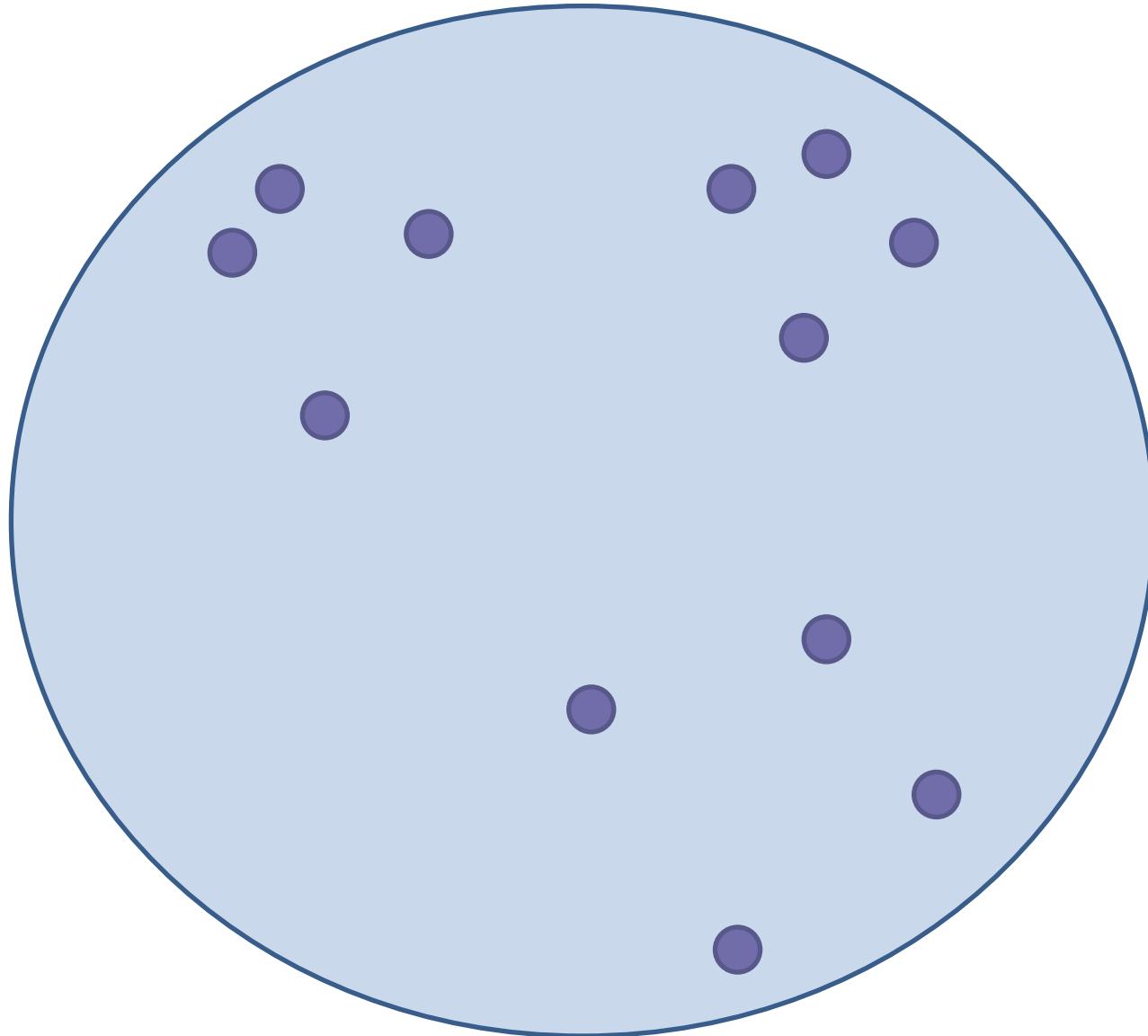
# Агломеративная кластеризация



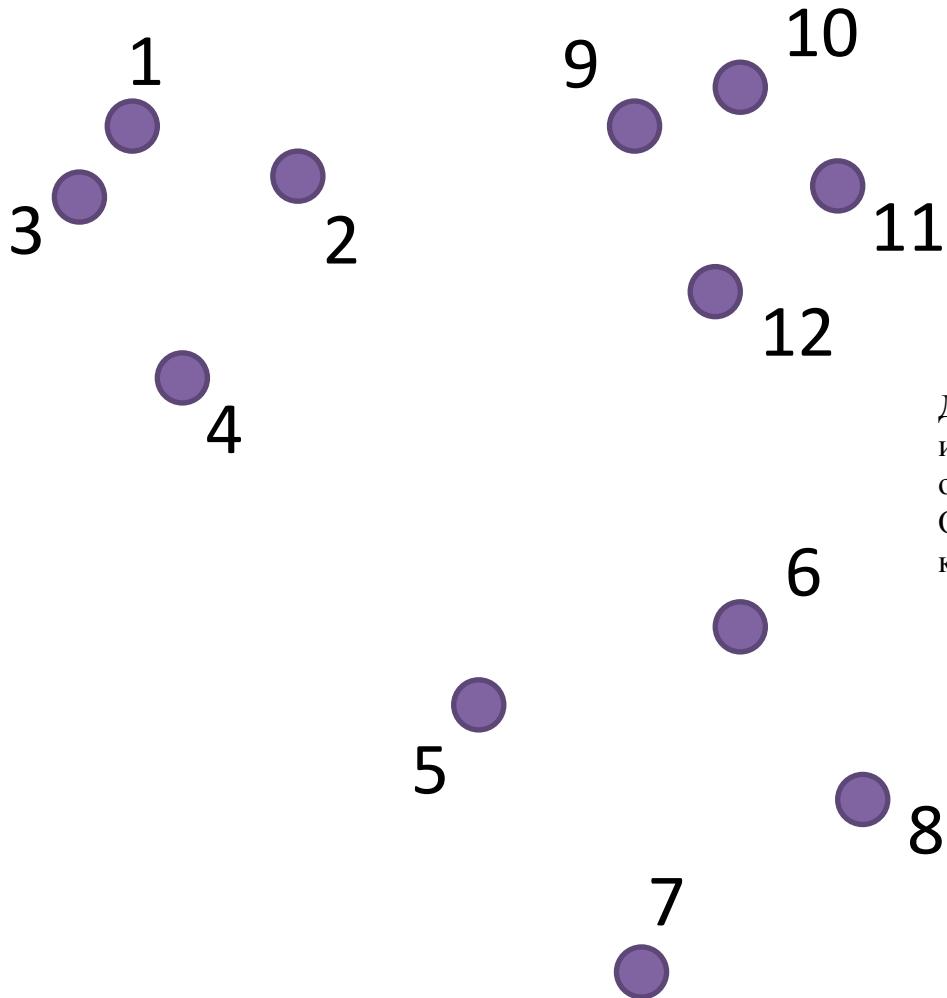
# Агломеративная кластеризация



# Агломеративная кластеризация

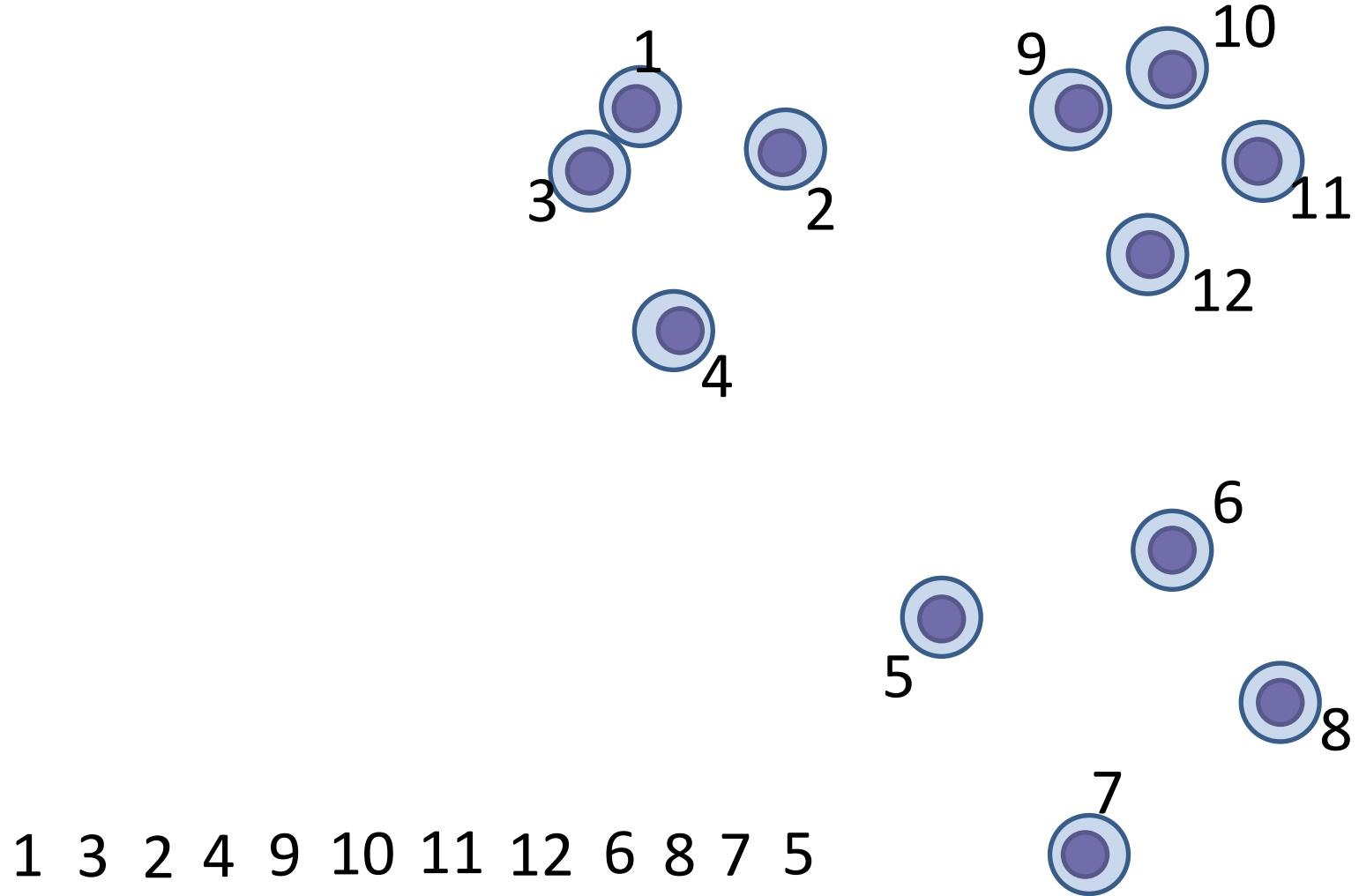


# Дендрограмма

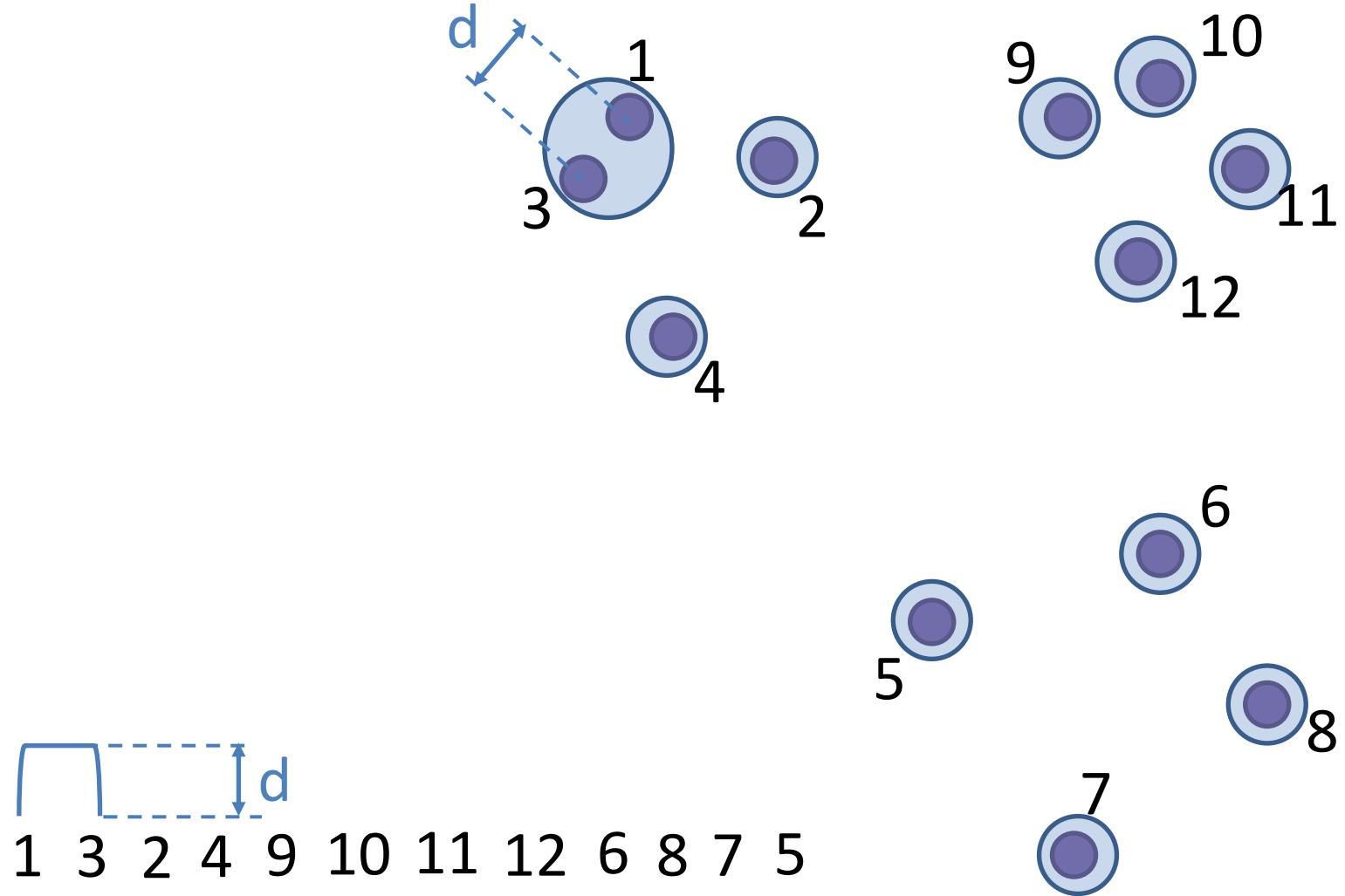


Дендрограмма — это графическое представление иерархической кластеризации, которое показывает, как объекты группируются в кластеры на разных уровнях. Она используется для визуализации результатов кластеризации и помогает понять структуру данных.

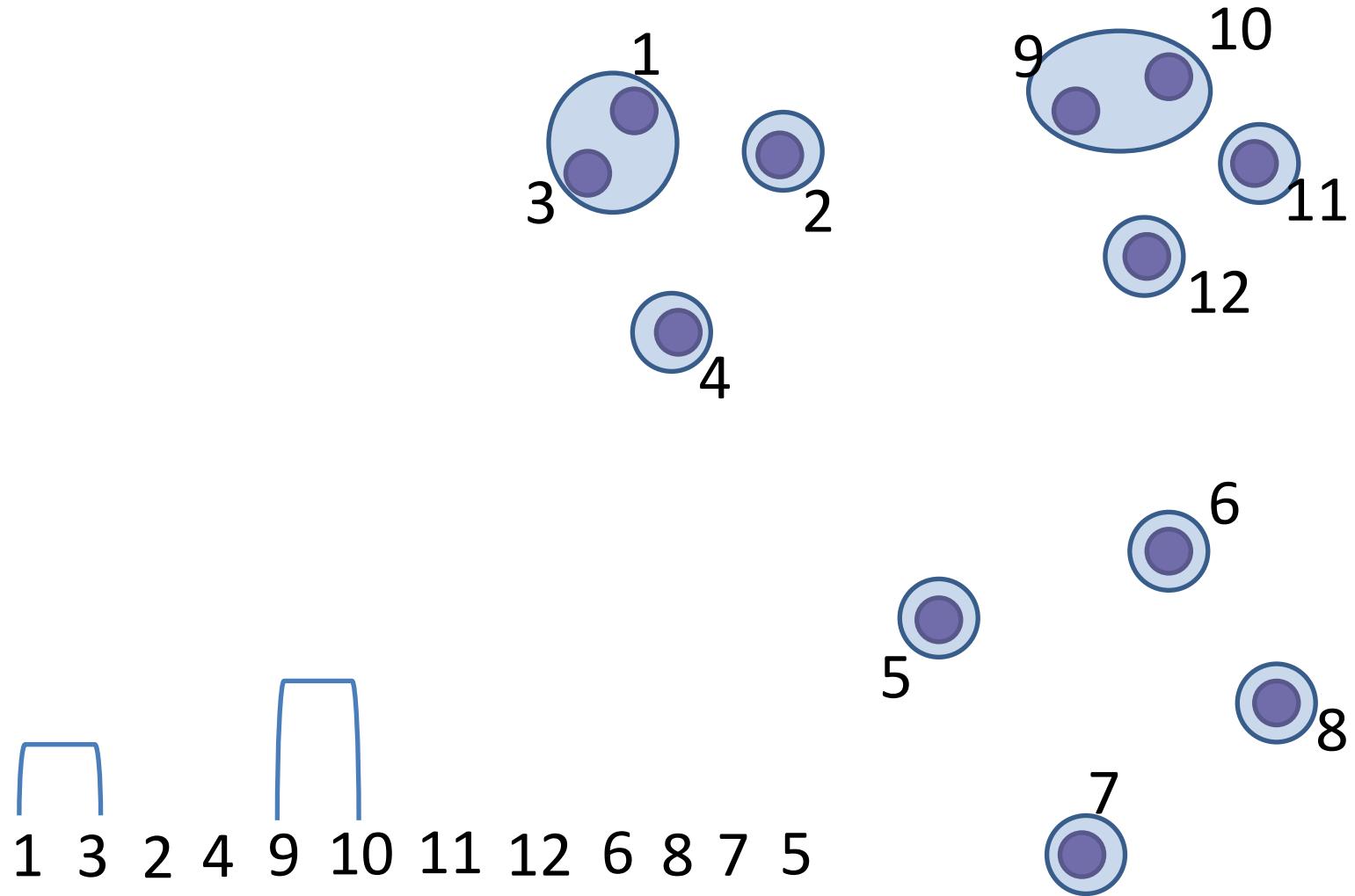
# Дендрограмма



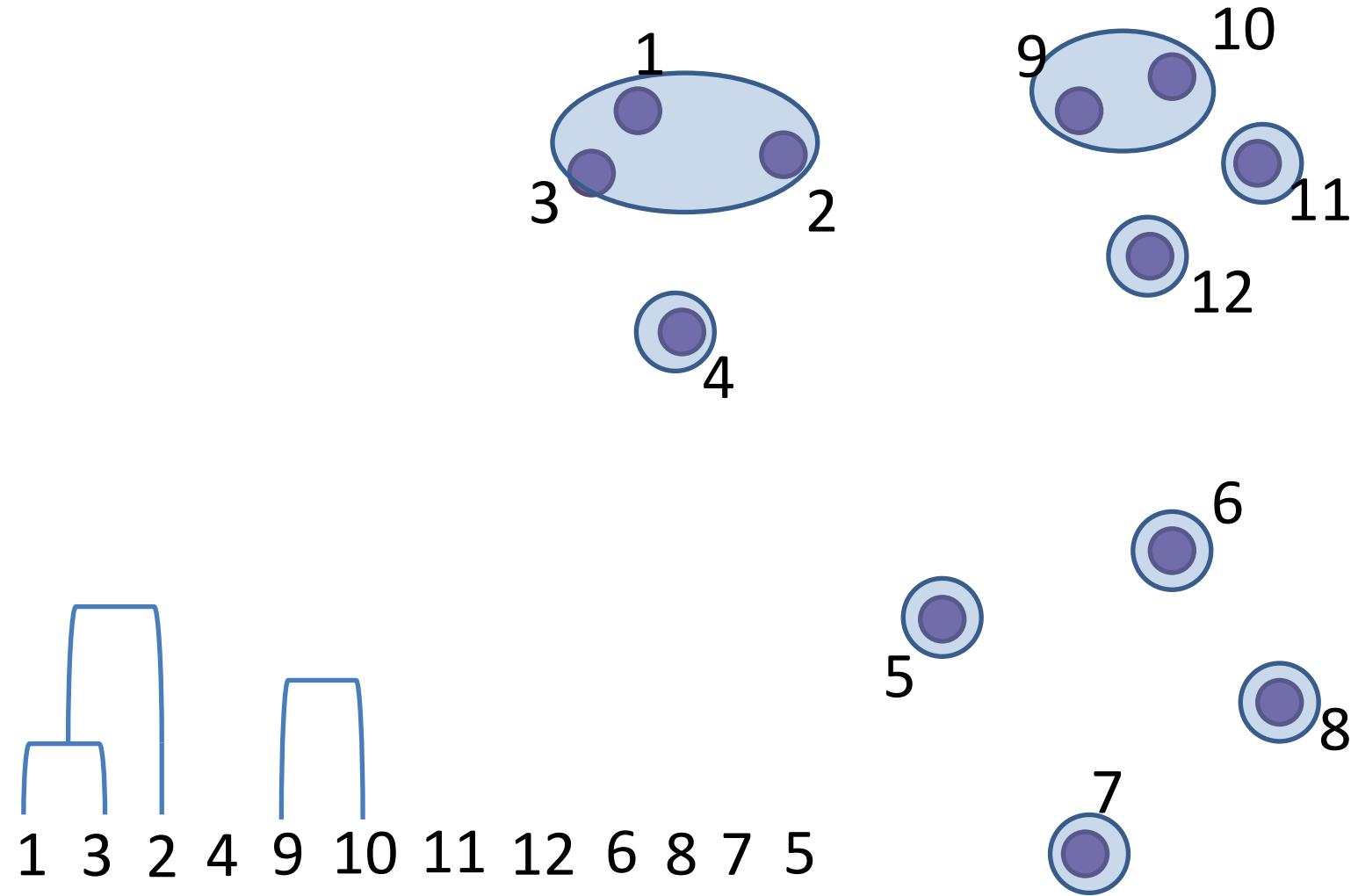
# Дендрограмма



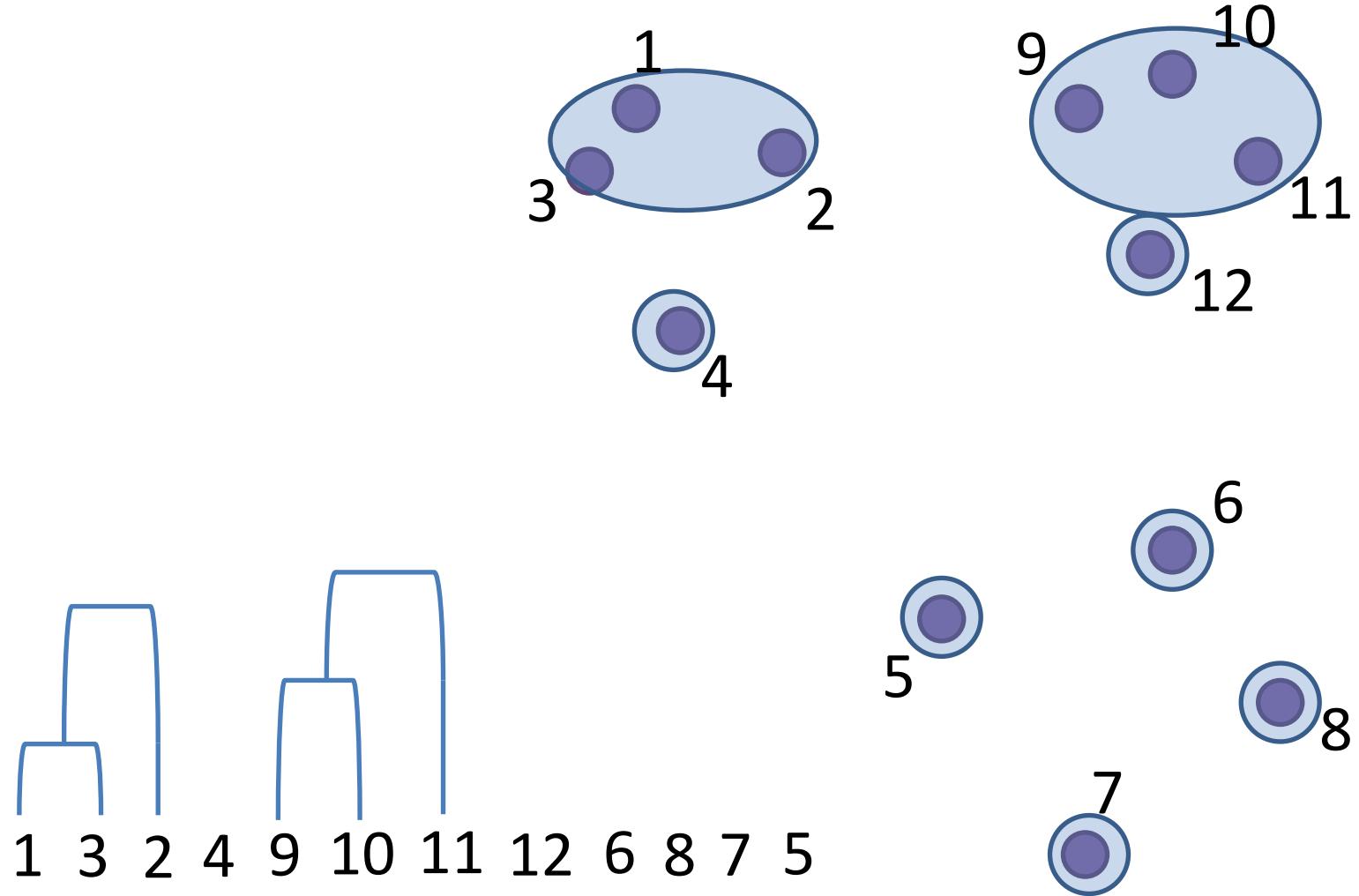
# Дендрограмма



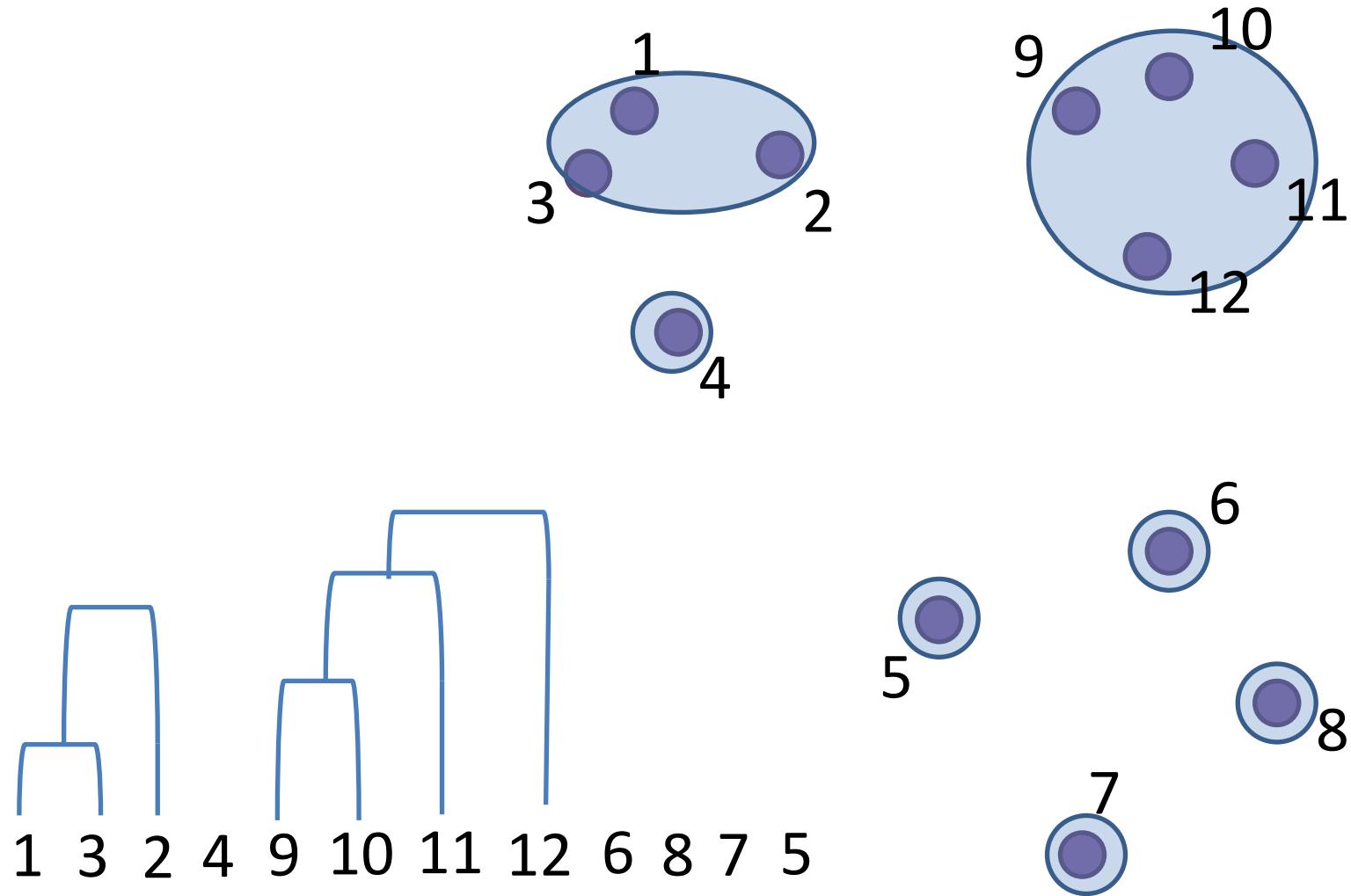
# Дендрограмма



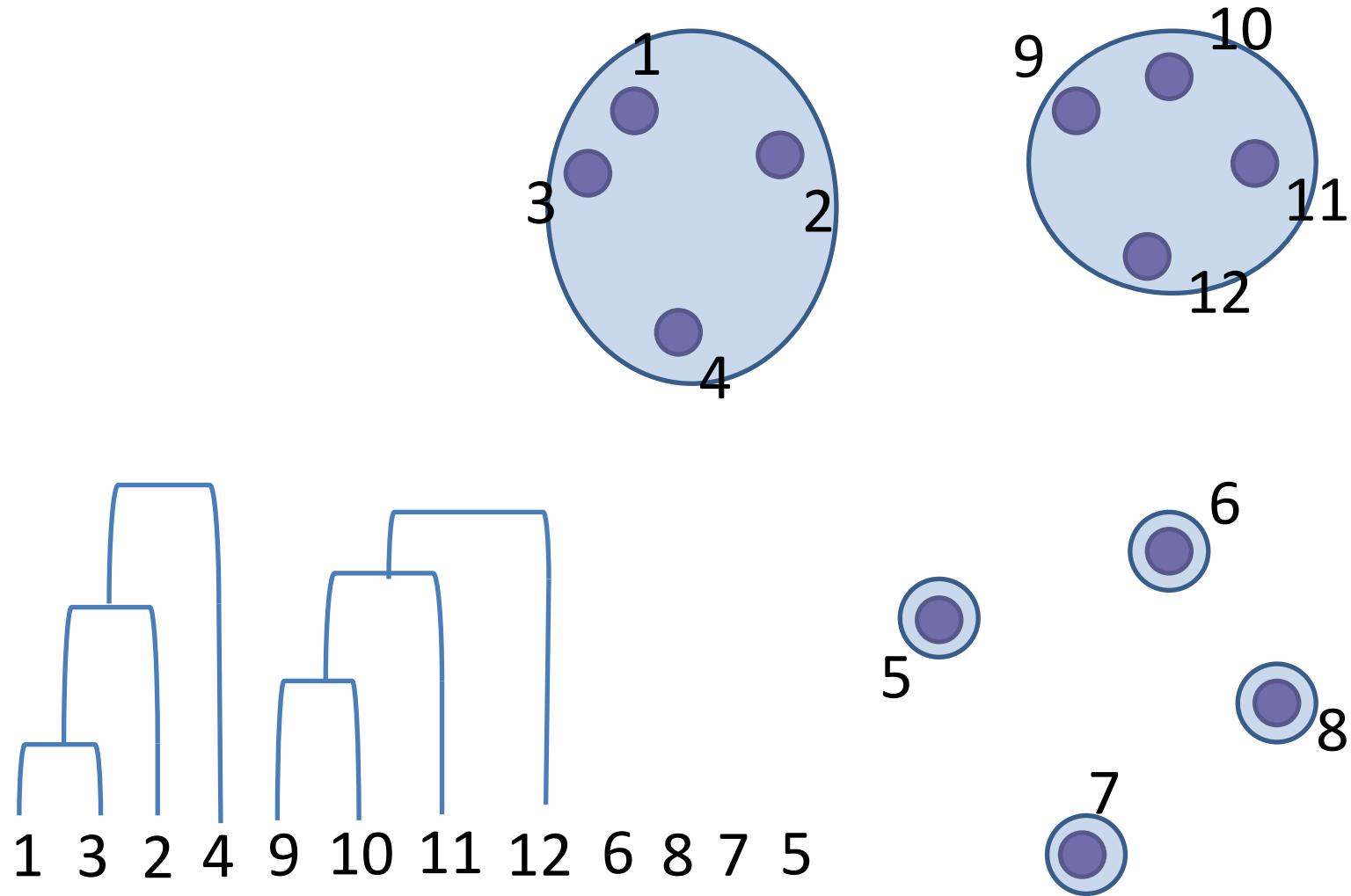
# Дендрограмма



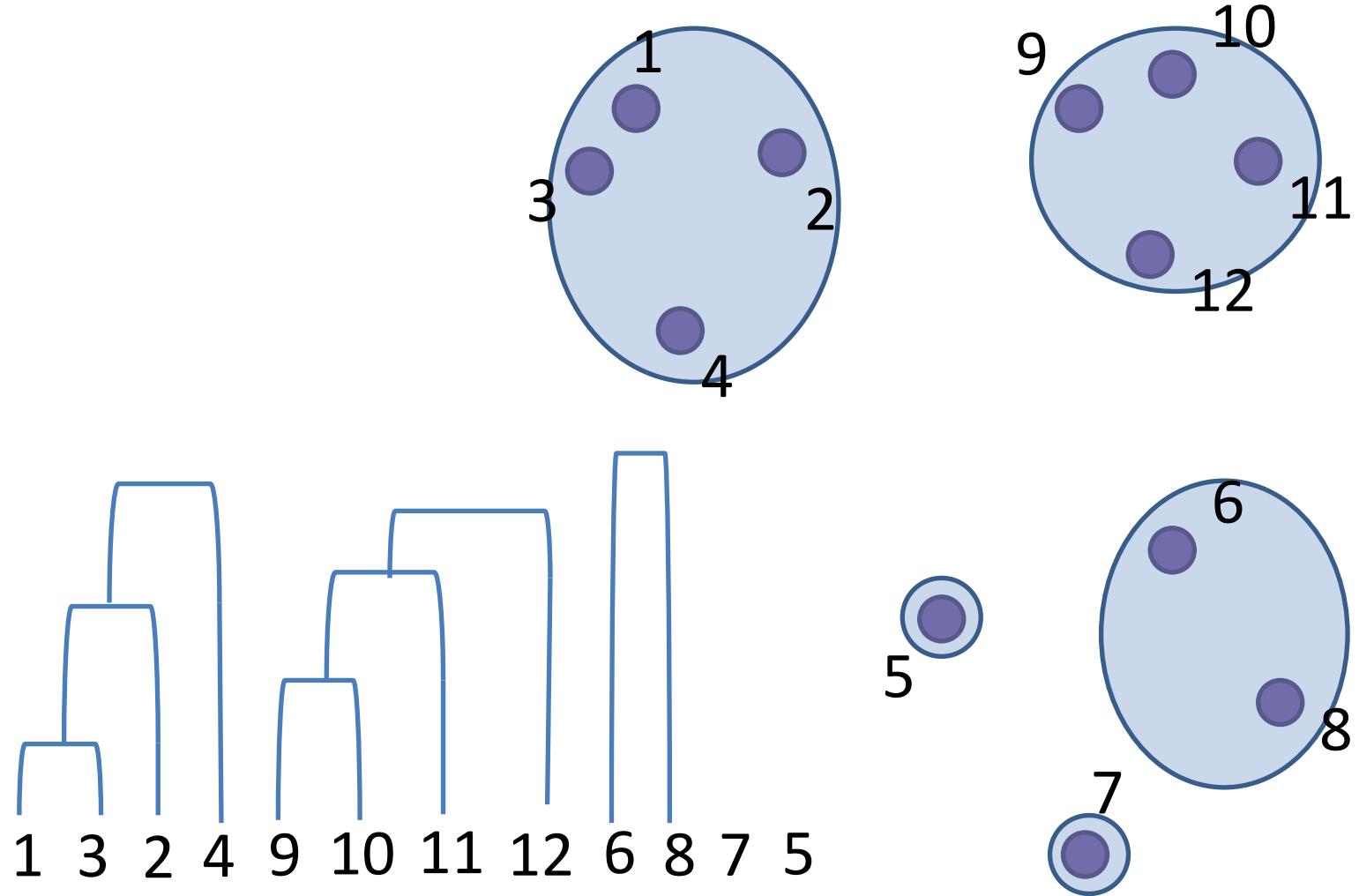
# Дендрограмма



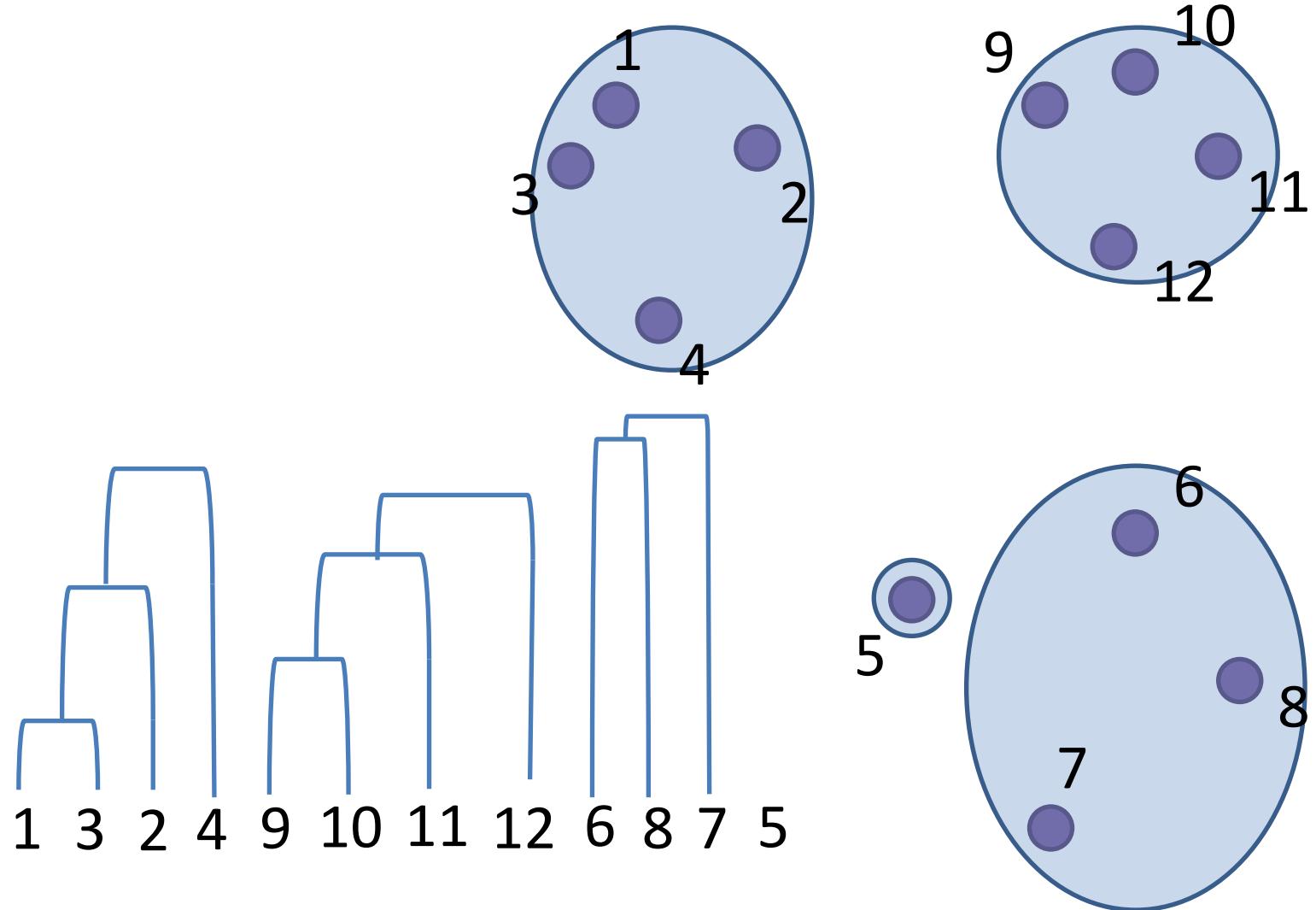
# Дендрограмма



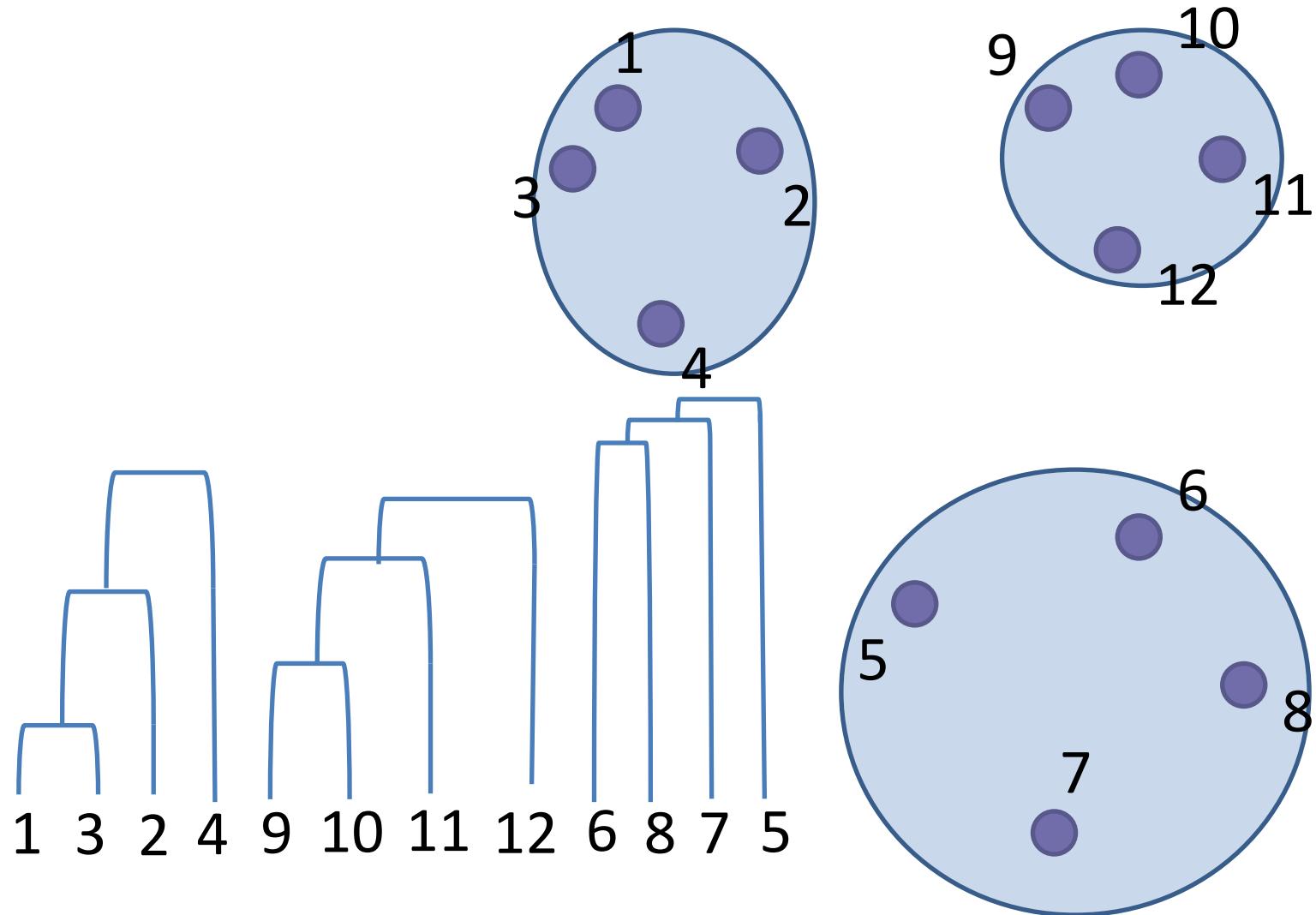
# Дендрограмма



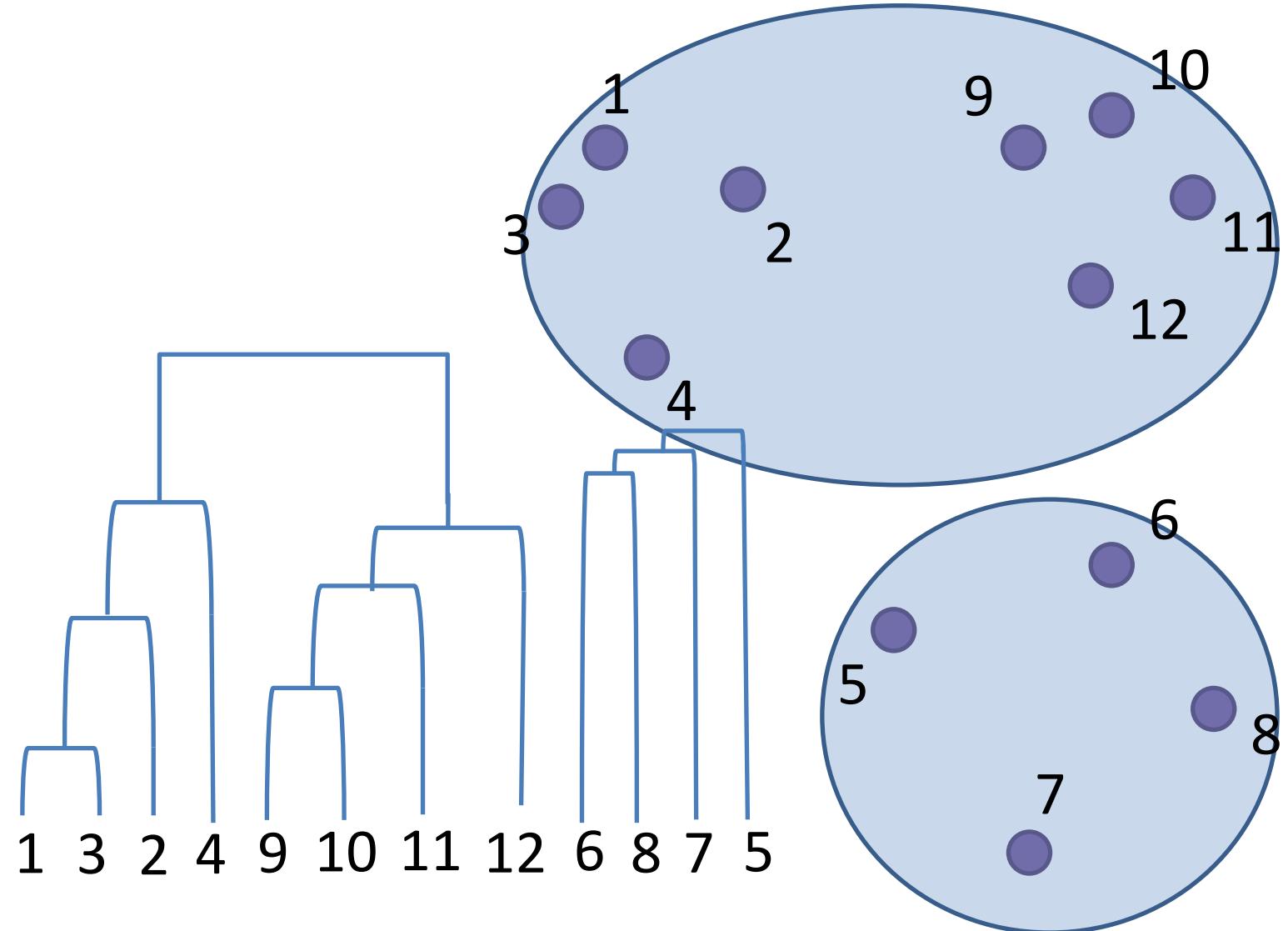
# Дендрограмма



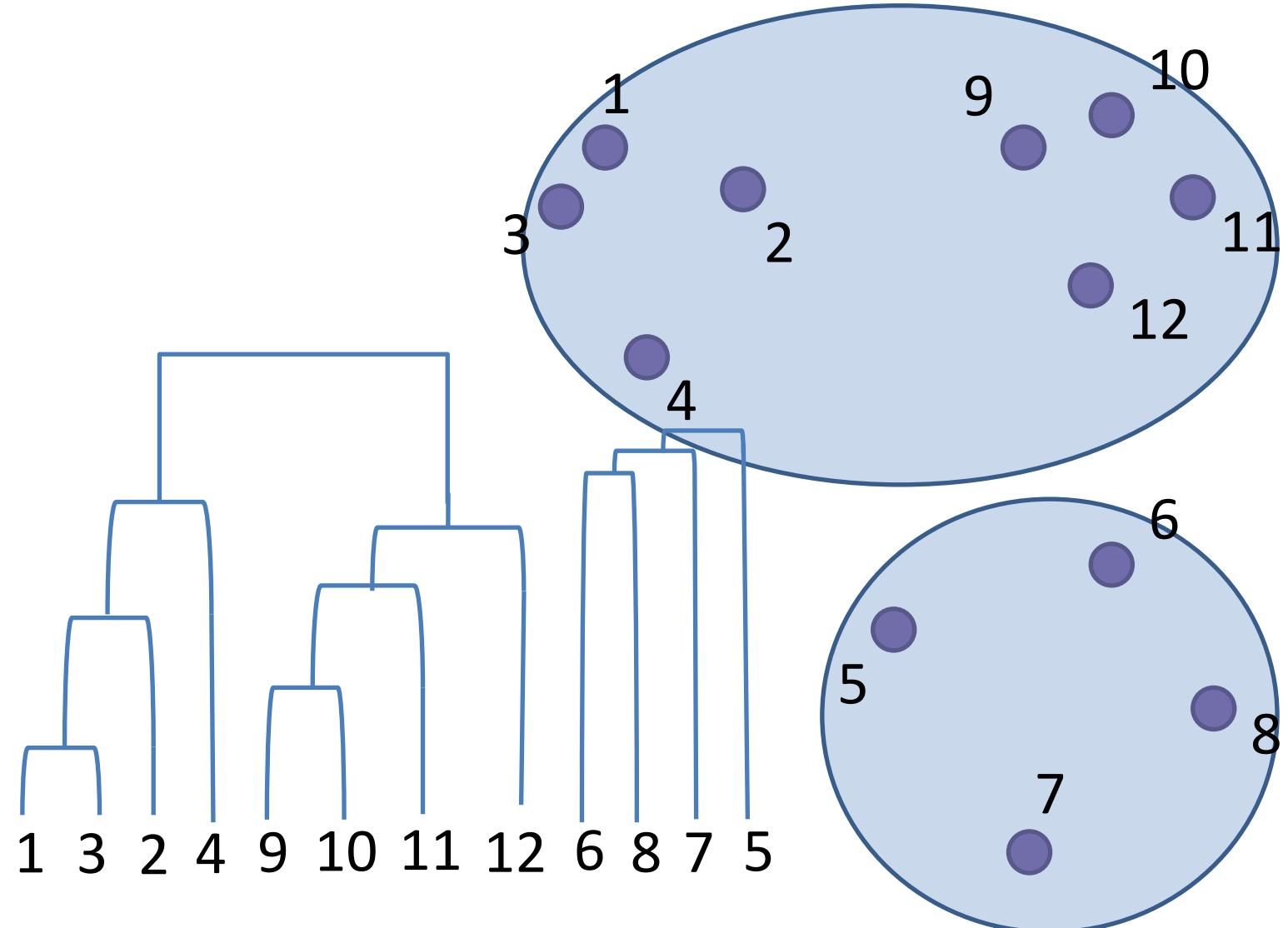
# Дендрограмма



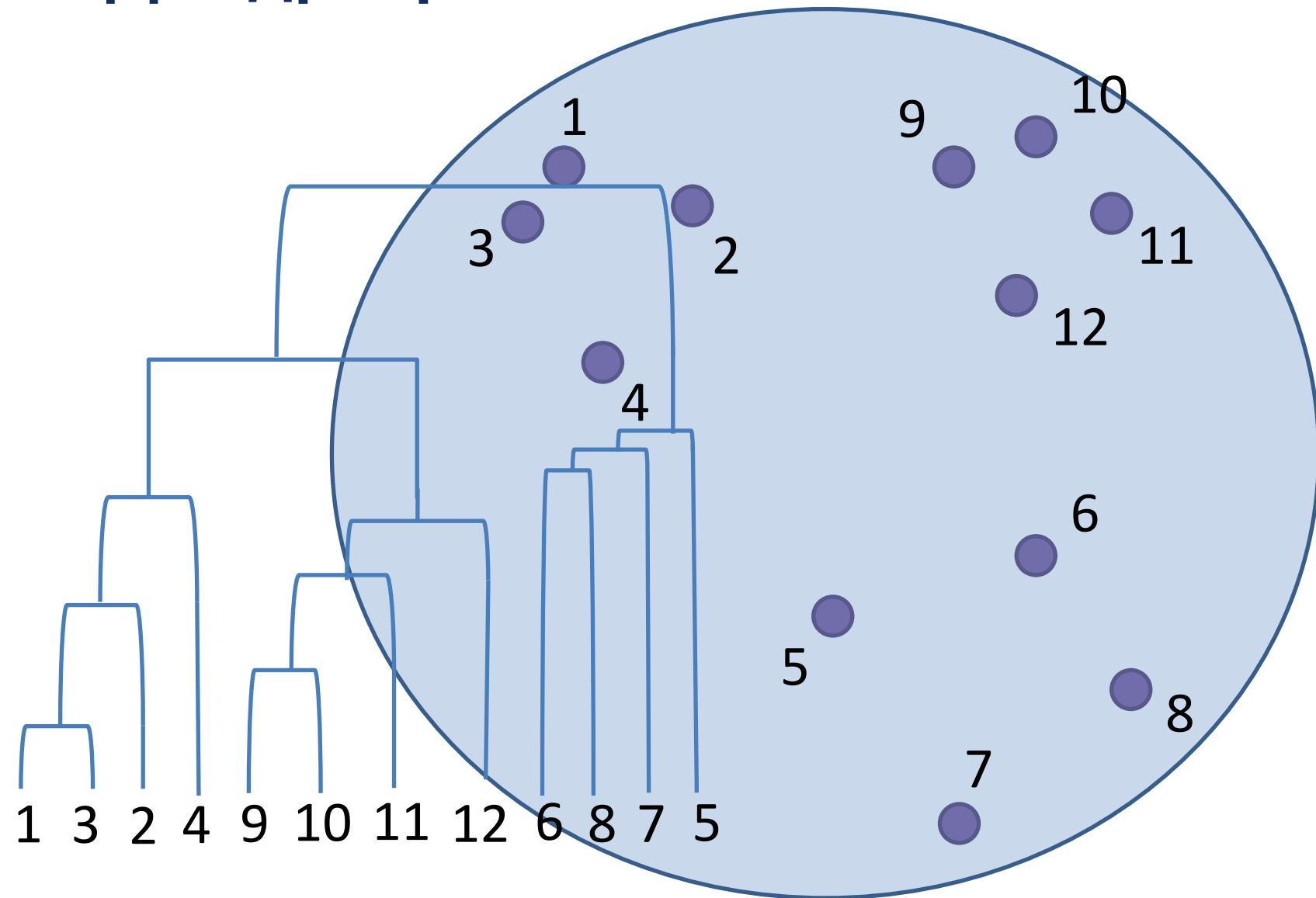
# Дендрограмма



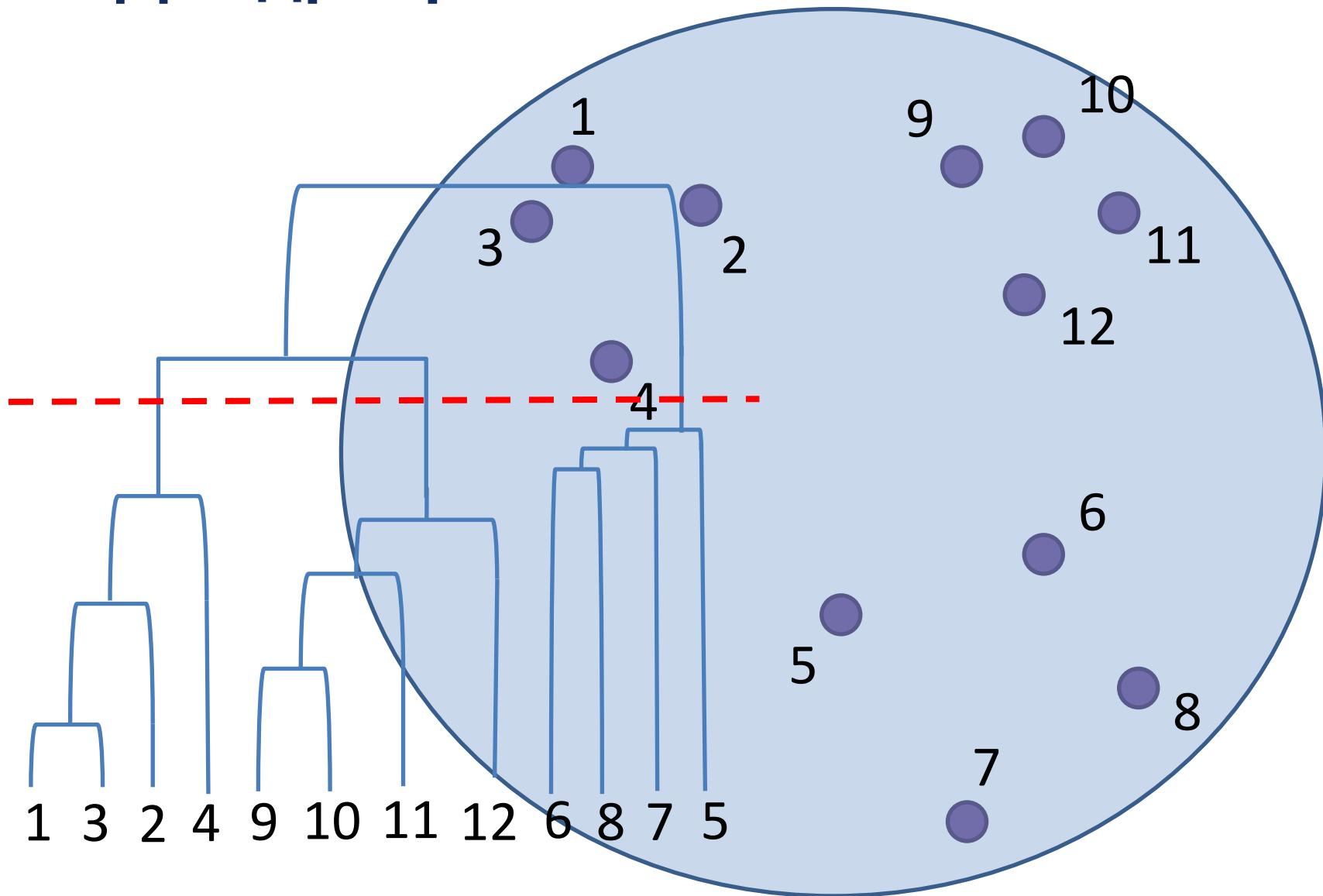
# Дендрограмма



# Дендрограмма



# Дендрограмма



# Агglomerативная кластеризация

1. Инициализация – каждая точка = кластер
2. Самые близкие (относительно какой-то метрики) кластеры объединяются
3. Повторяем до того момента, когда все точки будут в одном кластере
4. Останавливаемся, когда достигаем фиксированного числа кластеров, либо когда расстояние между кластерами больше заданного порога

# Резюме

- Кластеризация — задача без строгой постановки и без строгих критериев качества
- Много разновидностей в подходах
- Методы: K-Means, DBSCAN, иерархическая кластеризация и т.д.