Занятие 8. Кластеризация и визуализация данных.

Елена Кантонистова

elena.kantonistova@yandex.ru

ВШЭ, 2020

КЛАСТЕРИЗАЦИЯ

Даны объекты $x_1, \dots, x_l, x_i \in X$.

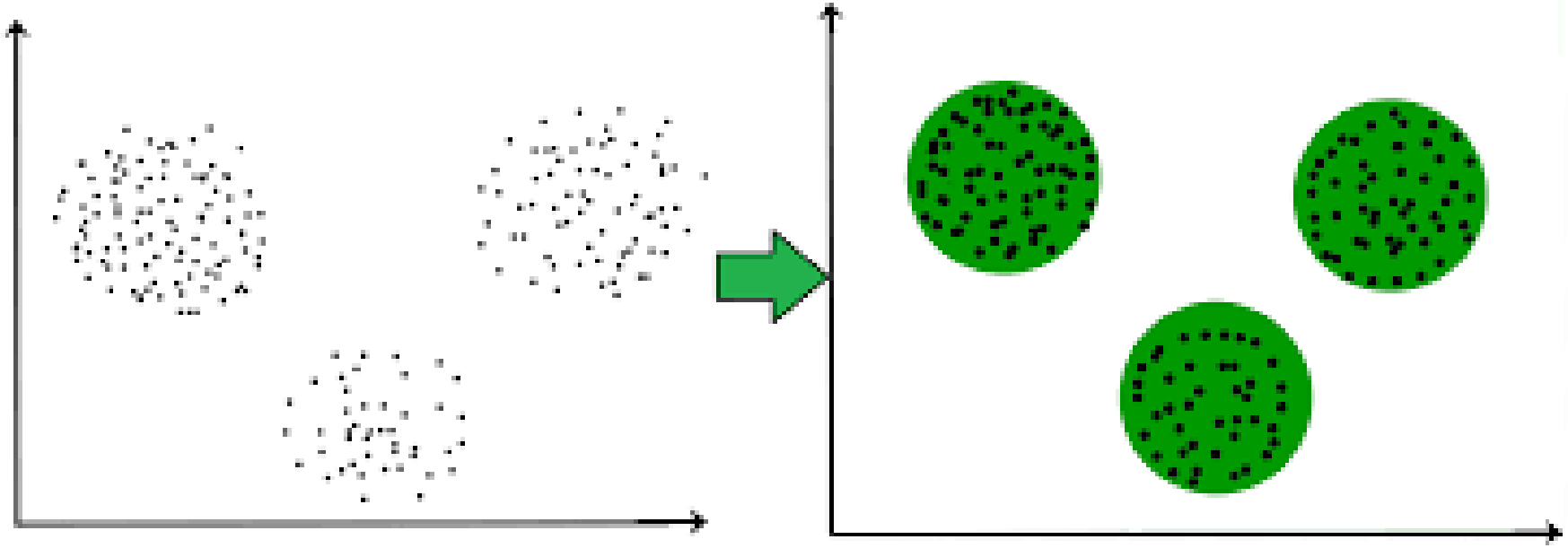
- Требуется выявить в данных K кластеров – таких областей, что объекты внутри одного кластера похожи друг на друга, а объекты из разных кластеров друг на друга не похожи.

КЛАСТЕРИЗАЦИЯ

Даны объекты $x_1, \dots, x_l, x_i \in X$.

- Требуется выявить в данных K кластеров – таких областей, что объекты внутри одного кластера похожи друг на друга, а объекты из разных кластеров друг на друга не похожи.
- Формализация задачи: необходимо построить алгоритм $a: X \to \{1, \dots, K\}$, сопоставляющий каждому объекту x номер кластера.

КЛАСТЕРИЗАЦИЯ



МЕТРИКИ КАЧЕСТВА КЛАСТЕРИЗАЦИИ

- ***Внешние метрики*** – используют информацию об истинных метках объектов
- ***Внутренние метрики*** – оценивают качество кластеризации, основываясь только на наборе данных.

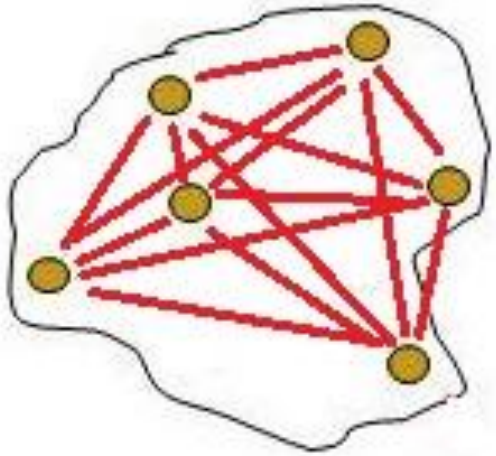
ВНУТРЕННИЕ МЕТРИКИ КАЧЕСТВА КЛАСТЕРИЗАЦИИ

ВНУТРИКЛАСТЕРНОЕ РАССТОЯНИЕ

Пусть c_k - центр k-го кластера

Внутри кластера все объекты максимально похожи, поэтому наша **цель – минимизировать внутрикластерное расстояние**:

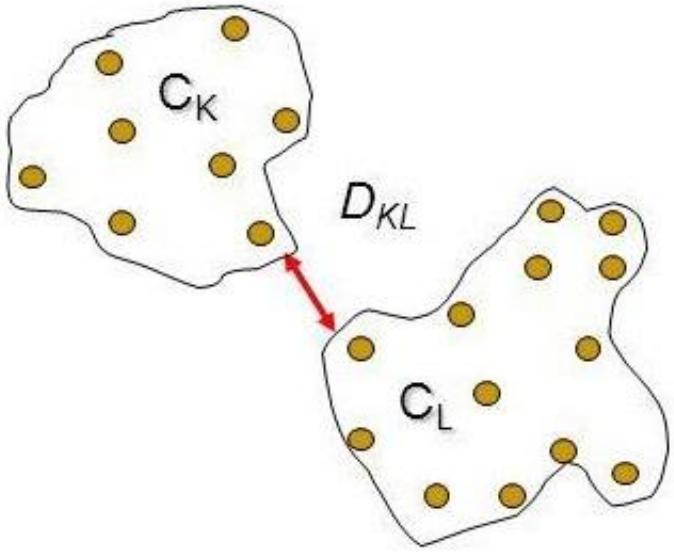
$$\sum_{k=1}^{K}\sum_{i=1}^{l}[a(x_i)=k]\rho(x_i,c_k)\to\min_a$$



МЕЖКЛАСТЕРНОЕ РАССТОЯНИЕ

Объекты из разных кластеров должны быть как можно менее похожи друг на друга, поэтому мы **максимизируем межкластерное расстояние**:

$$\sum_{i,j=1}^{l} [a(x_i) \neq a(x_j)]\rho(x_i, x_j) \to \max_a$$



ИНДЕКС ДАННА (DUNN INDEX)

Хотим **минимизировать внутрикластерное расстояние и одновременно максимизировать межкластерное расстояние**:

$$\frac{\min\limits_{1\le k<k'\le K} d(k,k')}{\max\limits_{1\le k\le K} d(k)} \to \max_a$$

$d(k,k')$ – расстояние между кластерами k и k',

$d(k)$ – внутрикластерное расстояние для k-го кластера.

ИНДЕКС ДАННА (DUNN INDEX)

Хотим **минимизировать внутрикластерное расстояние и одновременно максимизировать межкластерное расстояние**:

$$\frac{\min\limits_{1\le k<k'\le K} d(k,k')}{\max\limits_{1\le k\le K} d(k)} \to \max_a$$

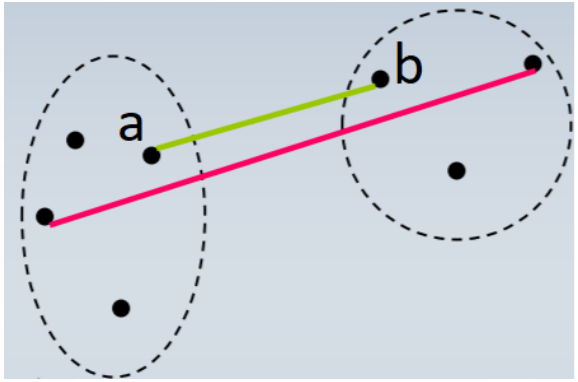
$d(k,k')$ – расстояние между кластерами k и k',

$d(k)$ – внутрикластерное расстояние для k-го кластера.

ВИДЫ РАССТОЯНИЙ МЕЖДУ ОБЪЕКТАМИ

- **Евклидово расстояние** – расстояние между точками в общепринятом понимании, то есть геометрическое расстояние между двумя точками.

$$\rho(a,b) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$



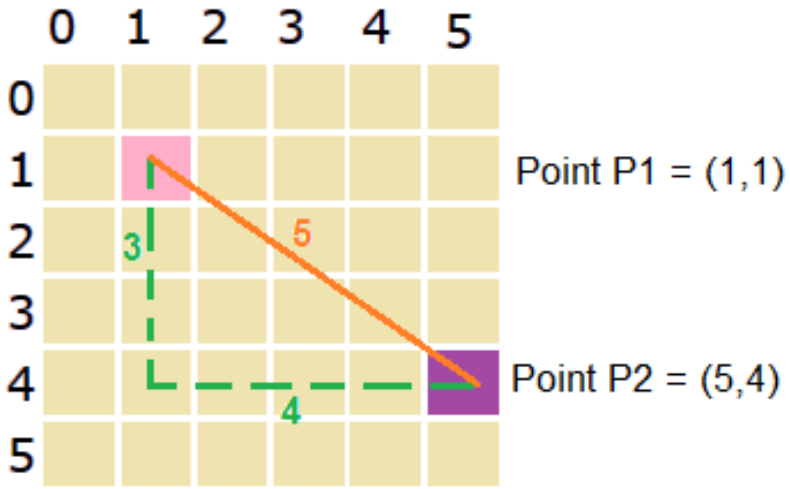
ВИДЫ РАССТОЯНИЙ МЕЖДУ ОБЪЕКТАМИ

- **Евклидово расстояние** – расстояние между точками в общепринятом понимании, то есть геометрическое расстояние между двумя точками.

$$\rho(a,b) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

- **Манхеттенское расстояние** (расстояние городских кварталов):

$$\rho(a,b) = |x_1 - x_2| + |y_1 - y_2|$$



Euclidean distance = $\sqrt{(5-1)^2 + (4-1)^2} = 5$

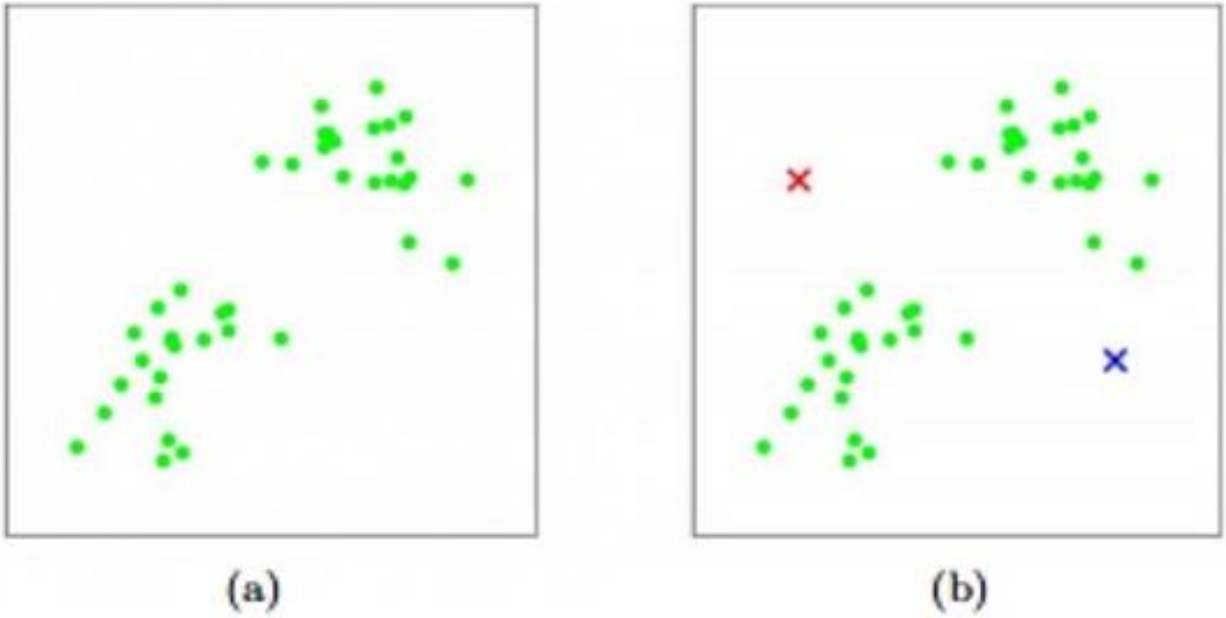
Manhattan distance = $|5-1| + |4-1| = 7$

K-MEANS

Дано: выборка $x_1, \dots, x_l$

Параметр: число кластеров K

Начало: **случайно выбрать центры кластеров $c_1, \dots, c_K$**



(a) (b)

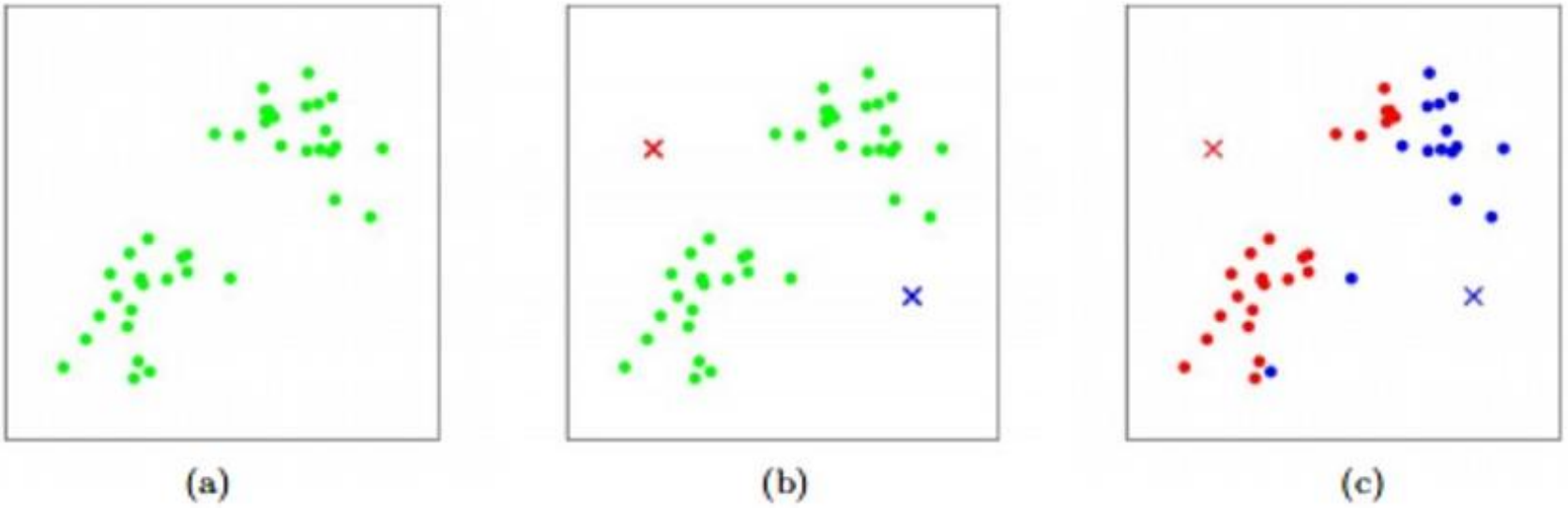
K-MEANS

Дано: выборка $x_1, \dots, x_l$

Параметр: число кластеров K

Начало: случайно выбрать центры кластеров $c_1, \dots, c_K$

1) **каждый объект отнести к ближайшему к нему центру кластера**



K-MEANS

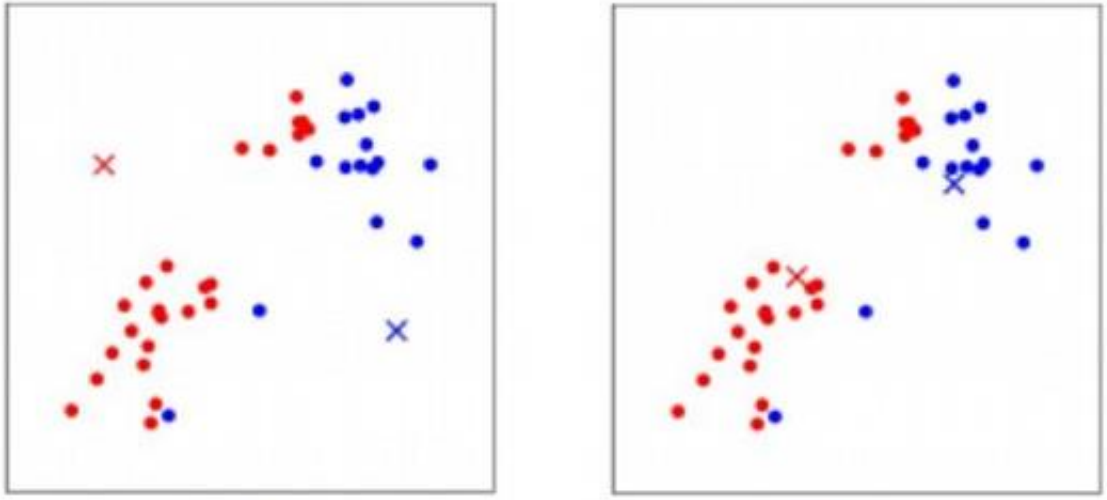
Дано: выборка $x_1, \dots, x_l$

Параметр: число кластеров K

Начало: случайно выбрать центры кластеров $c_1, \dots, c_K$

1) каждый объект отнести к ближайшему к нему центру кластера

2) пересчитать центры полученных кластеров



K-MEANS

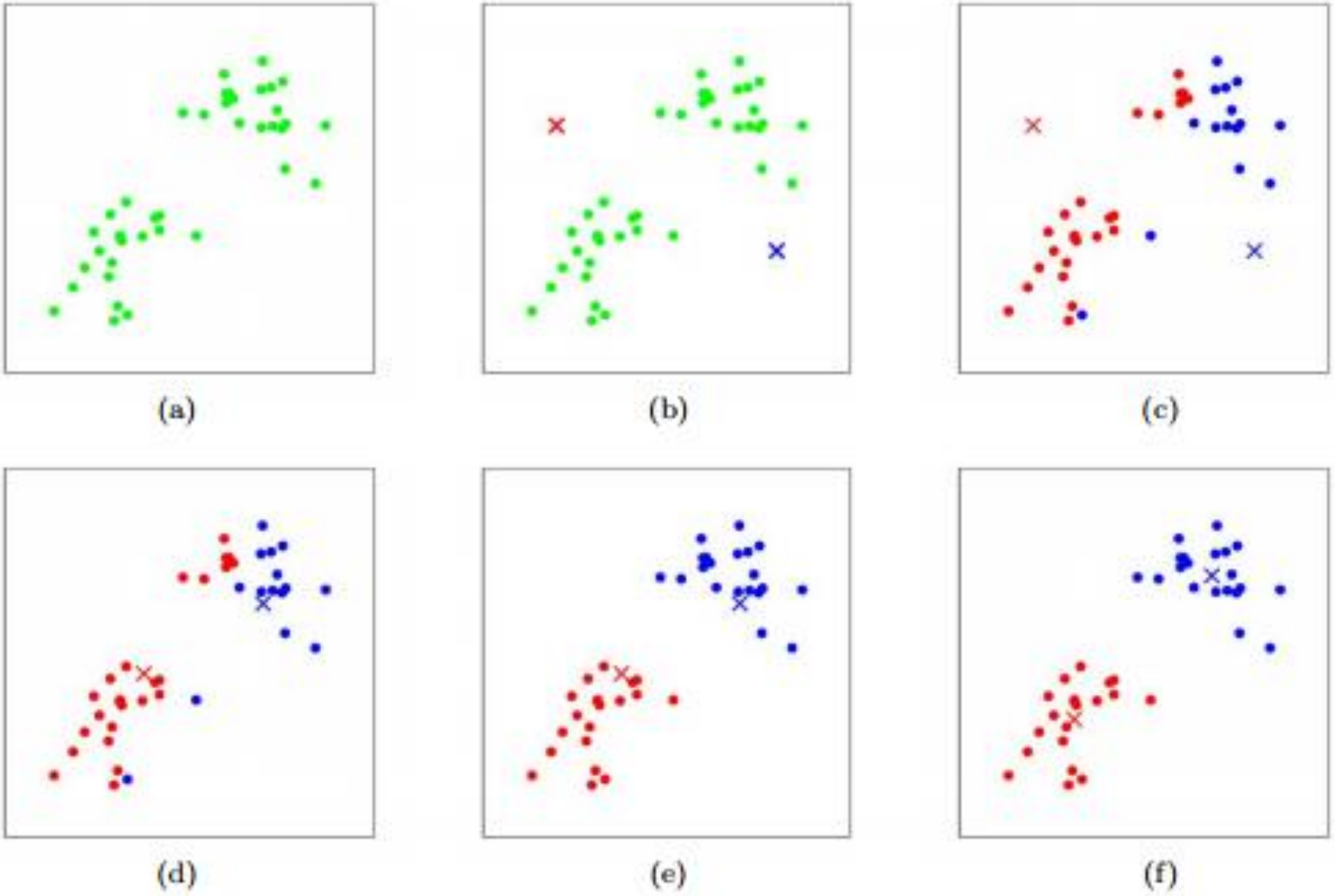
Дано: выборка $x_1, \ldots, x_l$

Параметр: число кластеров K

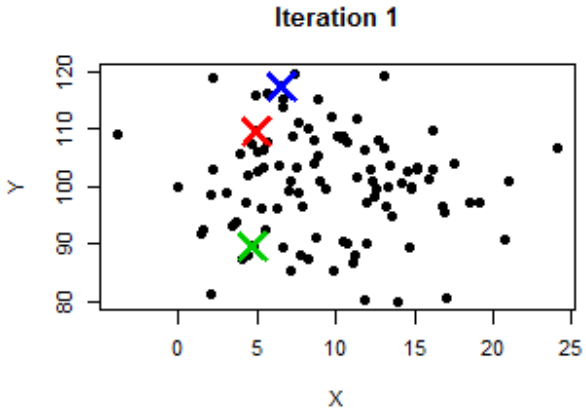
Начало: случайно выбрать центры кластеров $c_1, \ldots, c_K$

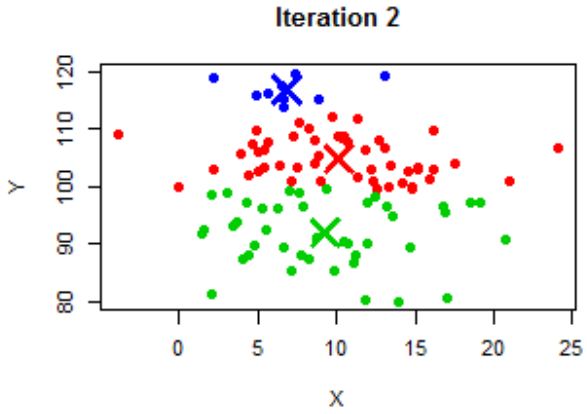
1) каждый объект отнести к ближайшему к нему центру кластера
2) пересчитать центры полученных кластеров
3) **повторить шаги 1 и 2 несколько раз до стабилизации кластеров**

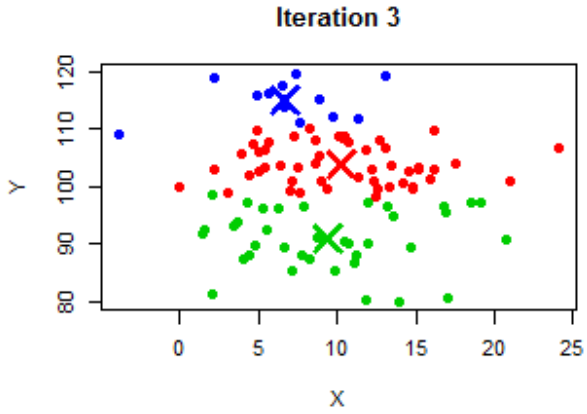
K-MEANS (ДВА КЛАСТЕРА)

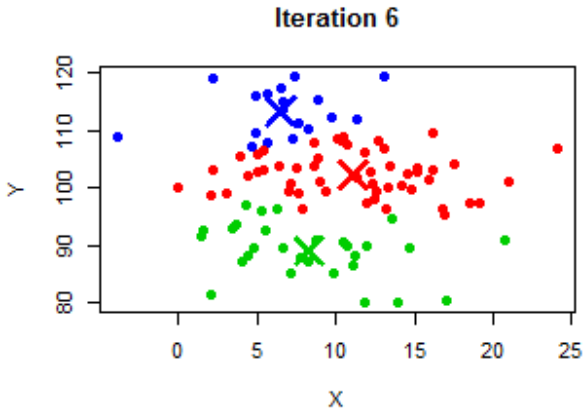


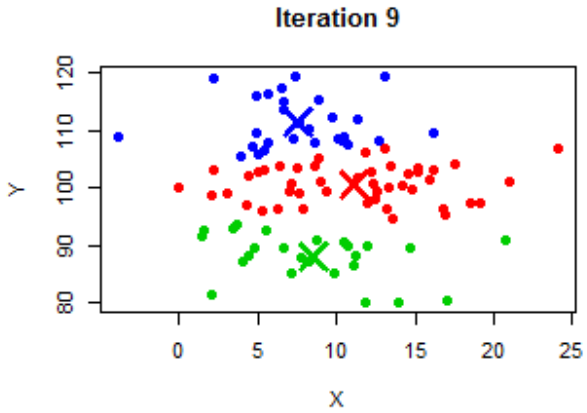
K-MEANS (ТРИ КЛАСТЕРА)

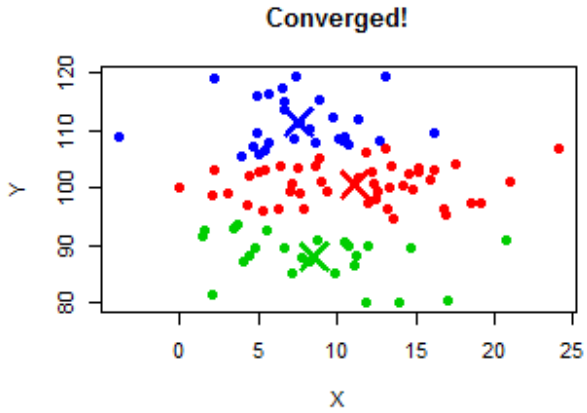








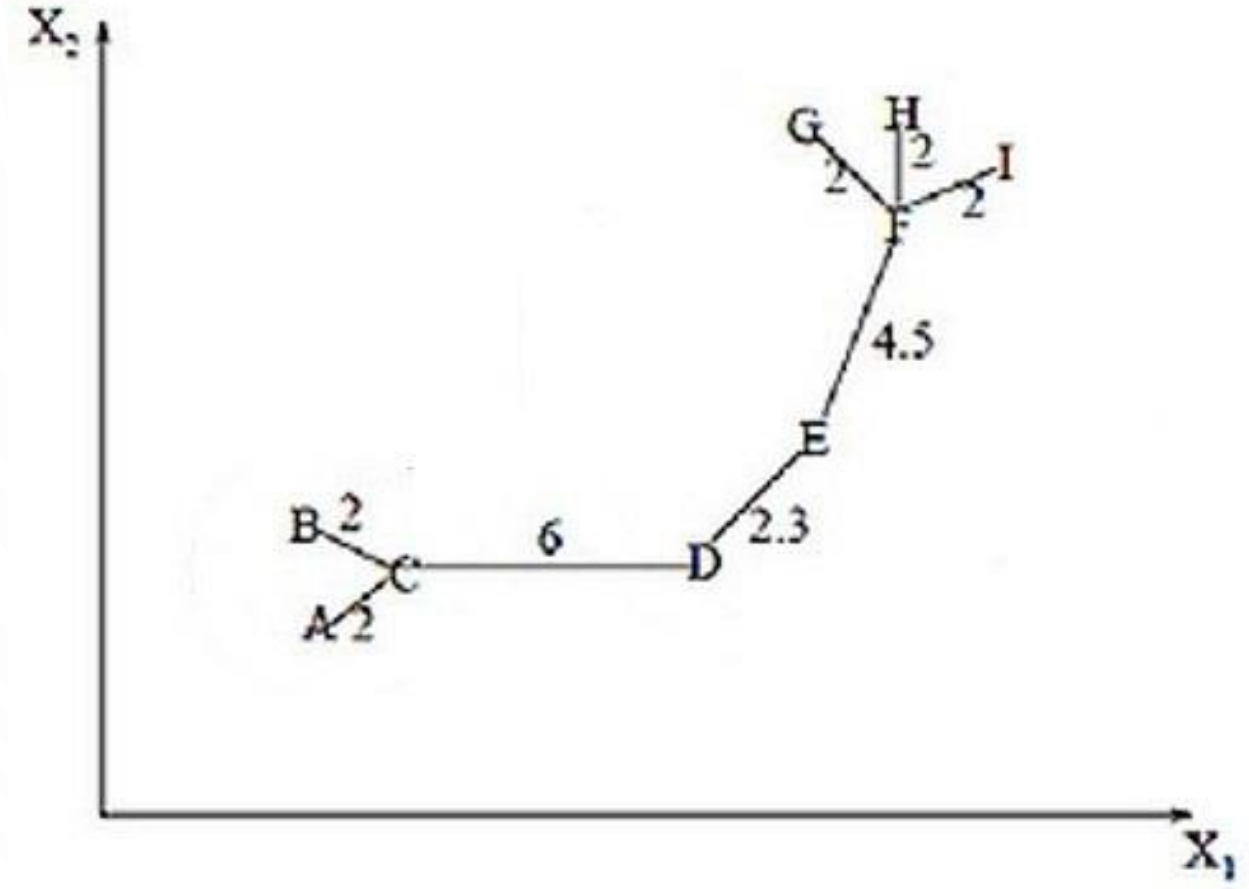




K-MEANS ДЛЯ СЖАТИЯ ИЗОБРАЖЕНИЙ

ГРАФОВЫЕ МЕТОДЫ КЛАСТЕРИЗАЦИИ

- выборка представляется в виде графа, где в вершинах стоят объекты, а на рёбрах – расстояния между ними



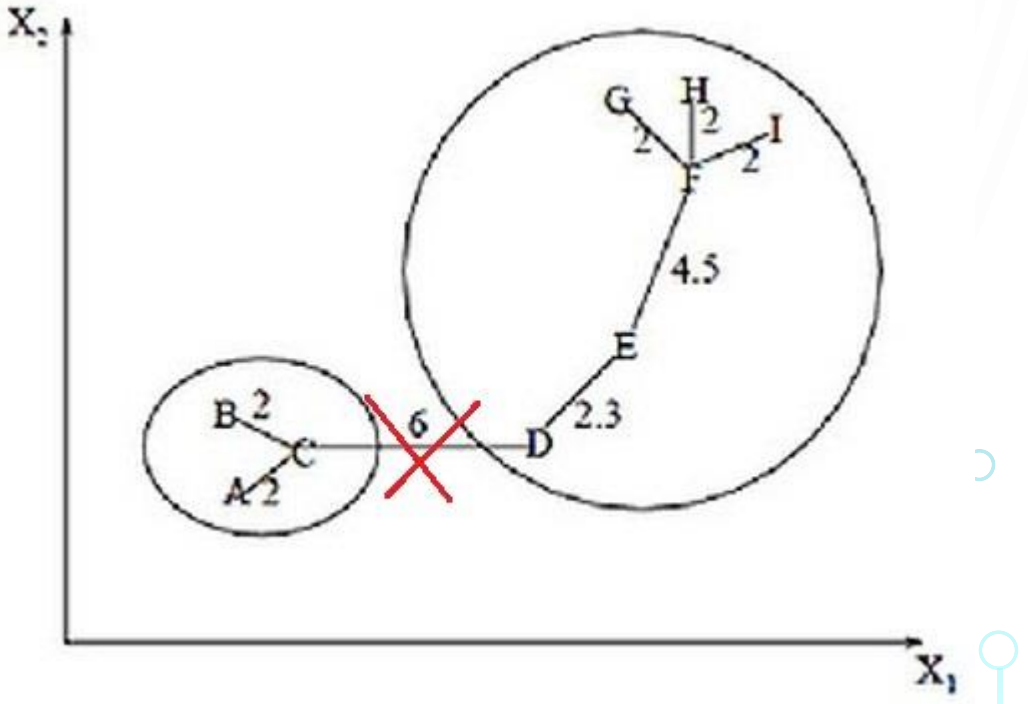
ГРАФОВЫЕ МЕТОДЫ КЛАСТЕРИЗАЦИИ

- выборка представляется в виде графа, где в вершинах стоят объекты, а на рёбрах – расстояния между ними

Алгоритм выделения связных компонент:

1) из графа удаляются все ребра, для которых расстояния больше некоторого значения R

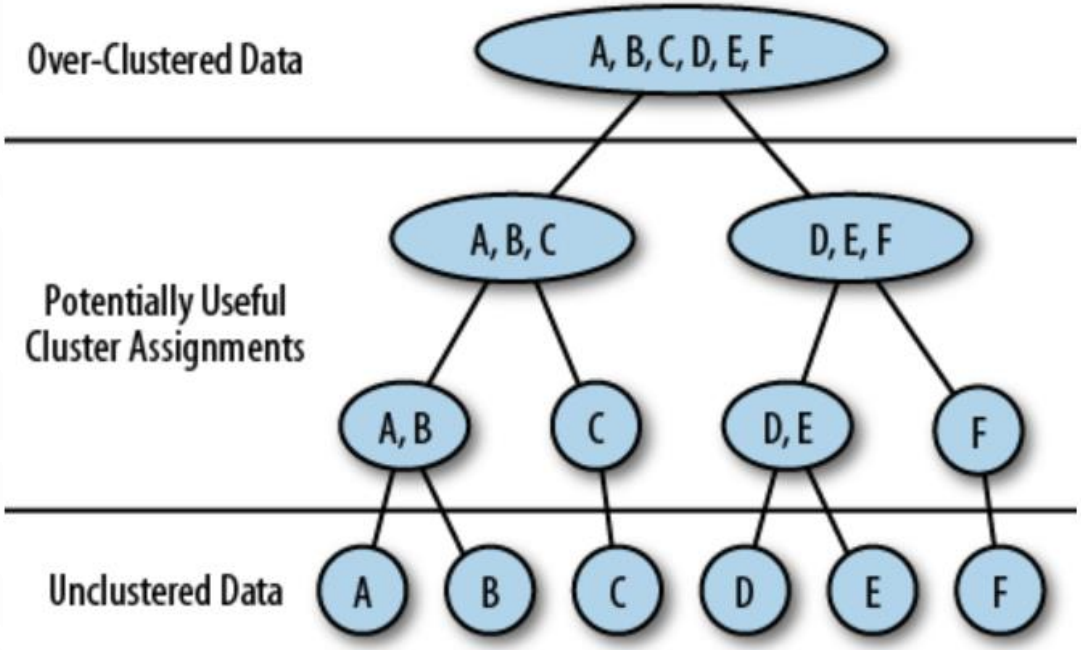
2) Кластеры – объекты, попадающие в одну компоненту связности



ИЕРАРХИЧЕСКАЯ КЛАСТЕРИЗАЦИЯ

Иерархия кластеров:

- на верхнем уровне – один большой кластер
- на нижнем уровне - l кластеров, каждый из которых состоит из одного объекта



ИЕРАРХИЧЕСКАЯ КЛАСТЕРИЗАЦИЯ

Алгоритм Ланса-Уильямса:

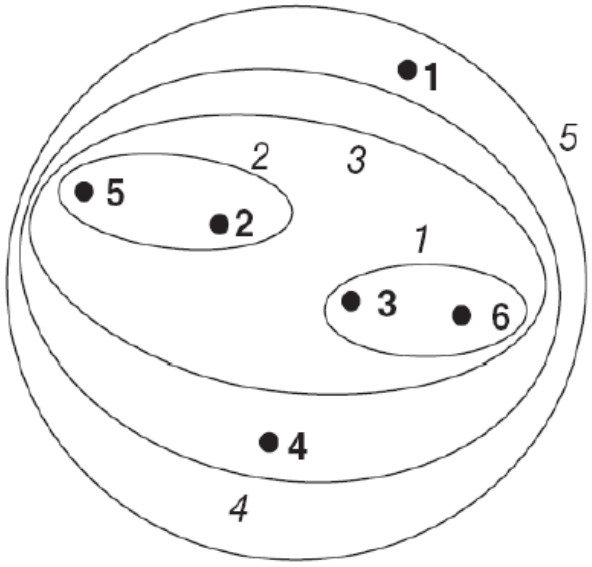
- первый шаг: один кластер = один объект
- на каждом следующем шаге объединяем два наиболее похожих кластера (по некоторой мере схожести d) с предыдущего шага

ИЕРАРХИЧЕСКАЯ КЛАСТЕРИЗАЦИЯ

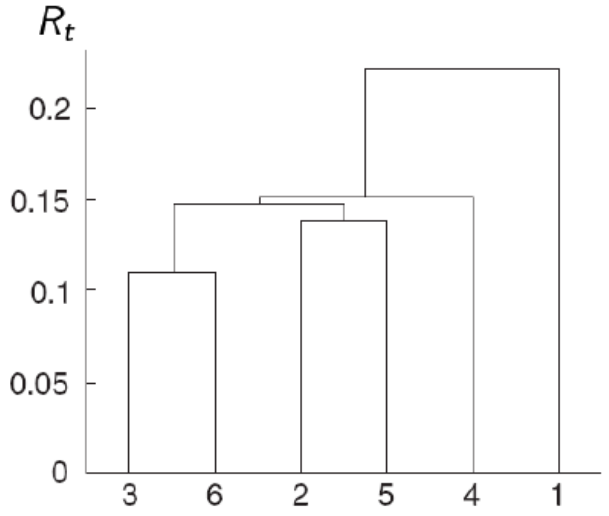
Алгоритм Ланса-Уильямса:

- первый шаг: один кластер = один объект
- на каждом следующем шаге объединяем два наиболее похожих кластера (по некоторой мере схожести d) с предыдущего шага

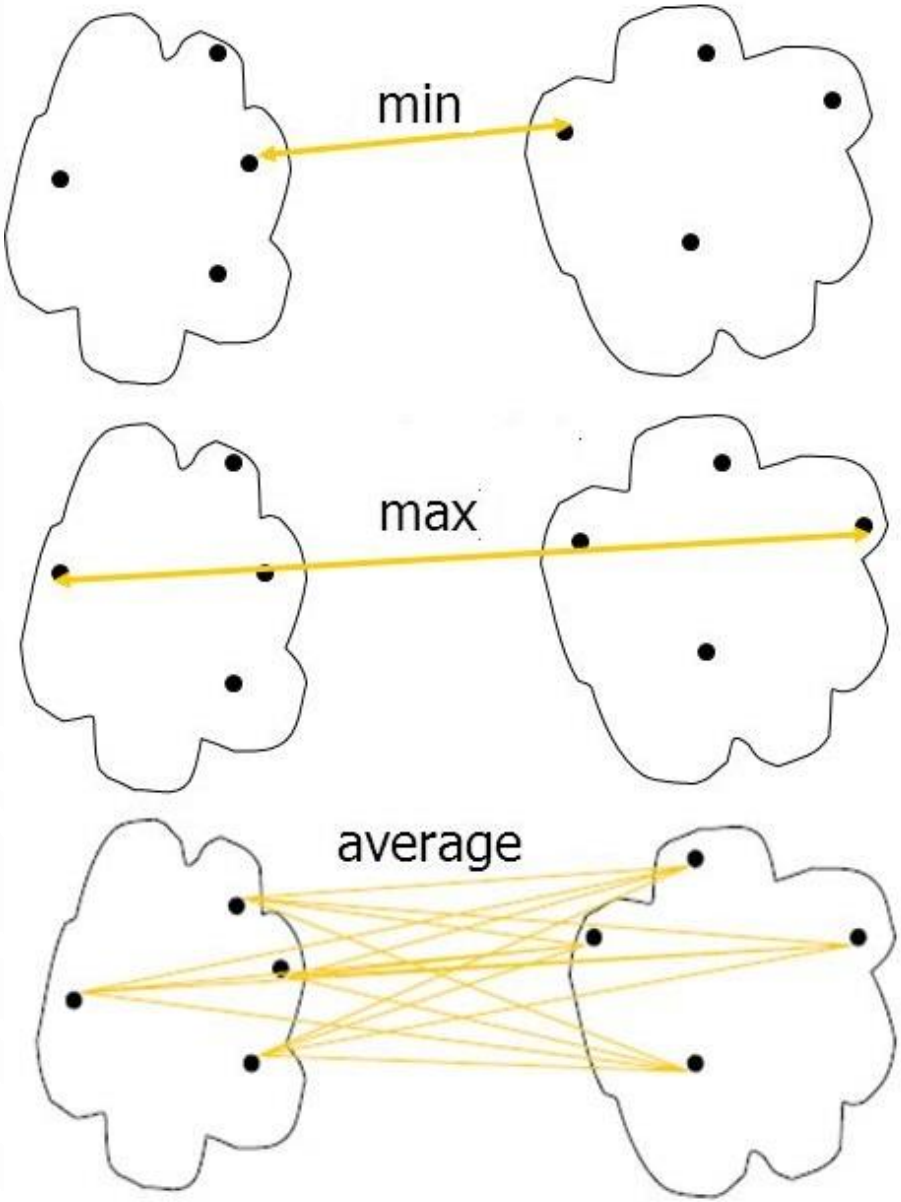
Диаграмма вложения



Дендрограмма

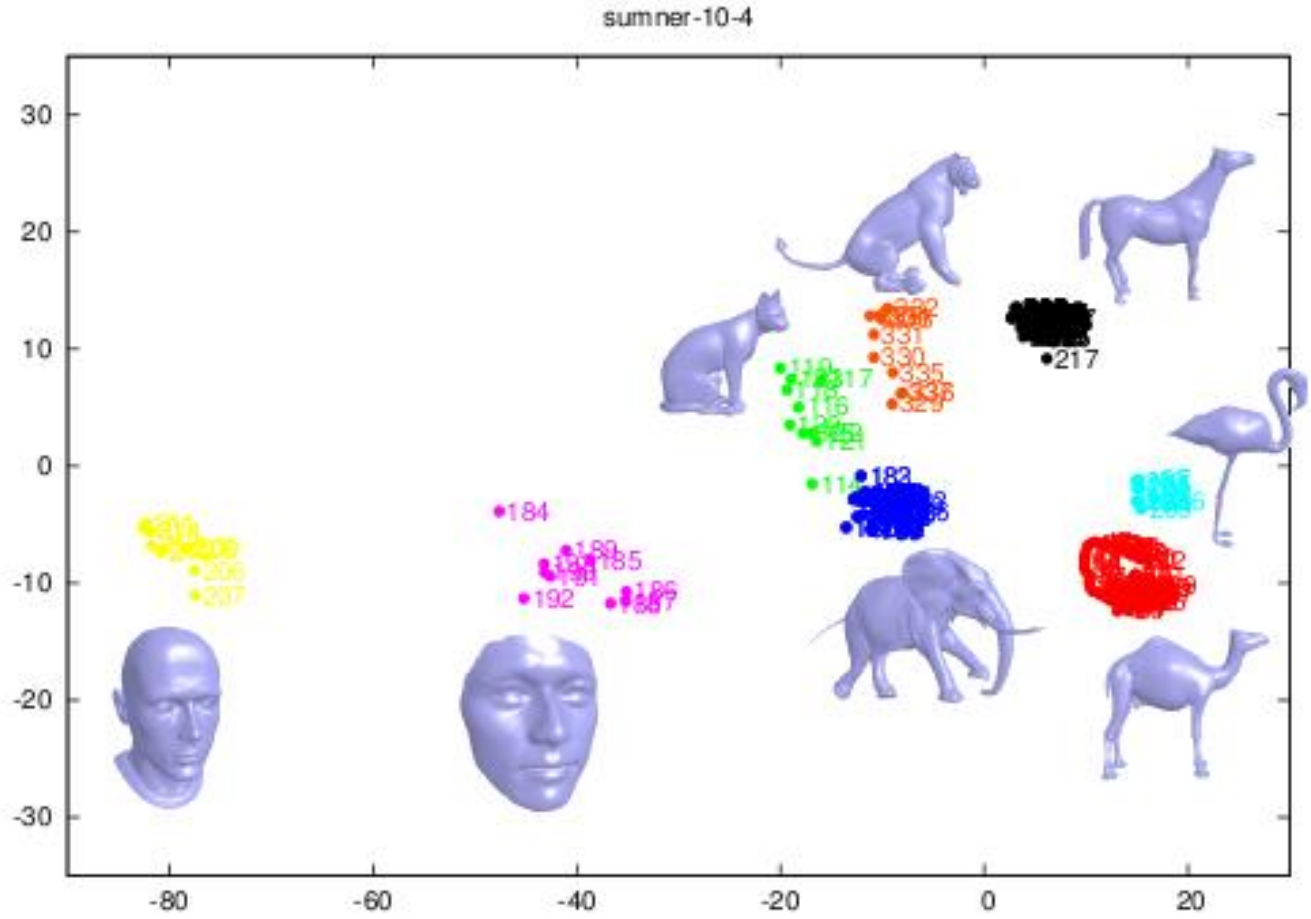


РАССТОЯНИЕ МЕЖДУ КЛАСТЕРАМИ



ВИЗУАЛИЗАЦИЯ

Задача визуализации состоит в отображении объектов в 2х- или 3хмерное пространство с сохранением отношений между ними.



MULTIDIMENSIONAL SCALING (MDS)

Идея метода – ***минимизация квадратов отклонений между исходными и новыми попарными расстояниями***:

$$\sum_{i \neq j}^{l} \left(\rho(x_i, x_j) - \rho(z_i, z_j)\right)^2 \rightarrow \min_{z_1, \ldots, z_l}$$

TSNE

t-SNE – t-distributed stochastic neighbor embedding

- *При проекции нам важно не сохранение расстояний между объектами, а сохранение пропорций:*

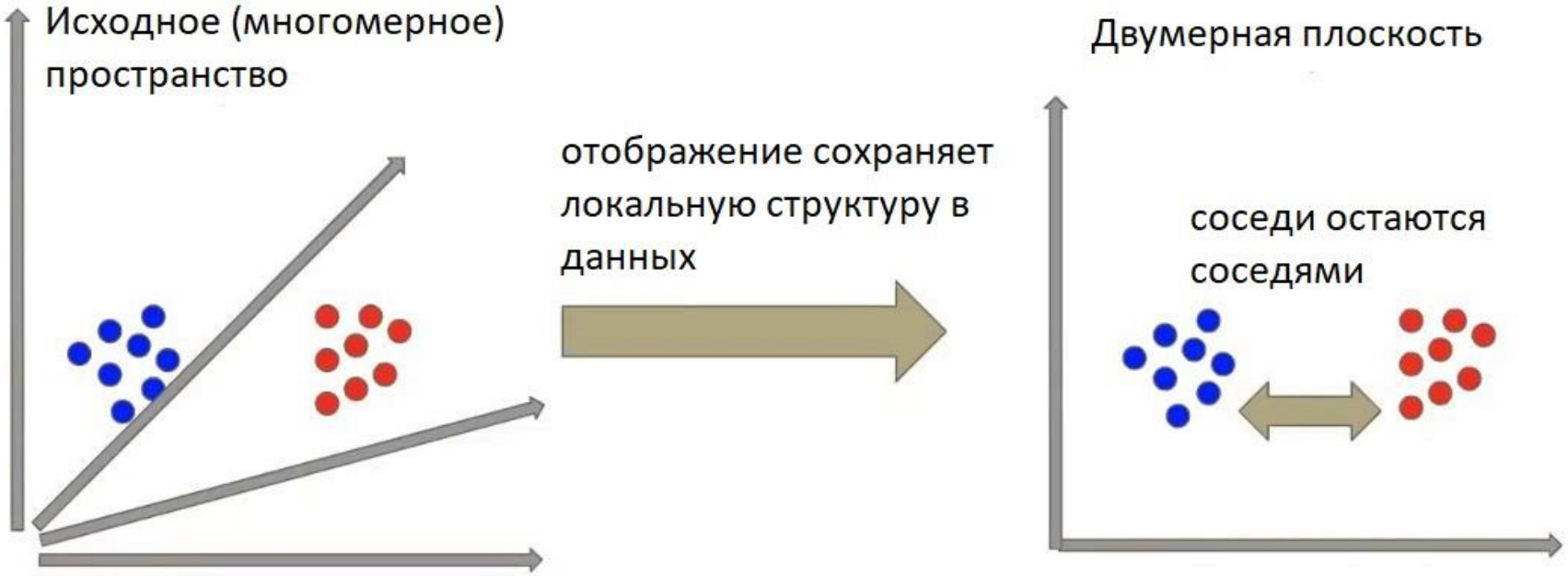
$$\rho(x_1, x_2) = \alpha\rho(x_1, x_3) \Rightarrow \rho(z_1, z_2) = \alpha\rho(z_1, z_3)$$

TSNE

t-SNE – t-distributed stochastic neighbor embedding

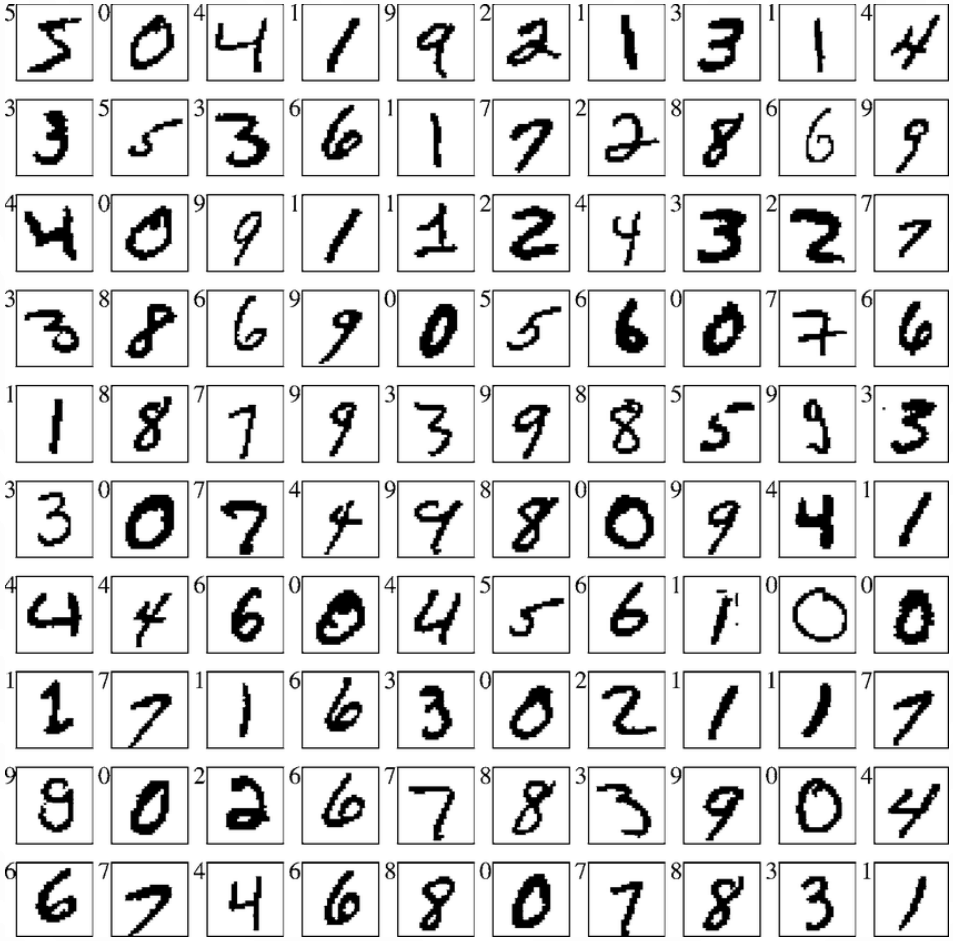
- *При проекции нам важно не сохранение расстояний между объектами, а сохранение пропорций:*

$$\rho(x_1, x_2) = \alpha\rho(x_1, x_3) \Rightarrow \rho(z_1, z_2) = \alpha\rho(z_1, z_3)$$

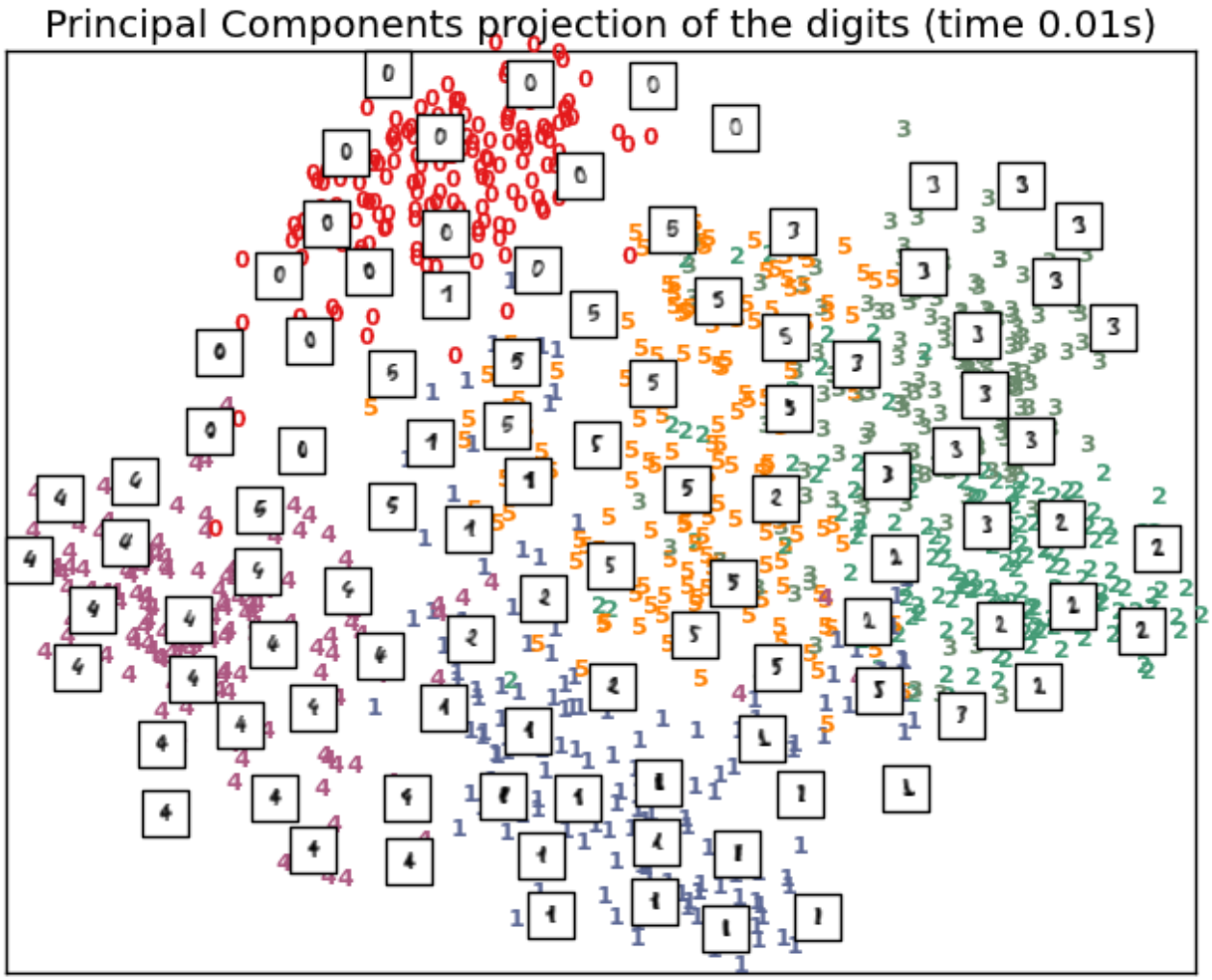


TSNE (ПРИМЕР)

- MNIST – датасет из различных написаний десятичных цифр, где каждая картинка размера 28x28.



PCA (ПРИМЕР)



TSNE (ПРИМЕР)