Введение в АД

Лекция 8

Кластеризация

Обучение без учителя

Обучение с учителем (supervised learning):

$x_1, ..., x_n$ — объекты.

$Y_1, ..., Y_n$ — таргет.

Требуется научиться предсказывать таргет по объектам.

- Задаем множество моделей и функционал ошибки.
- Обучение — выбор лучшей модели с точки зрения функционала.

Обучение без учителя (unsupervised learning):

$x_1, ..., x_n$ — объекты.

Отсутствует таргет.

Требуется исследовать данные на наличие внутренней структуры.

Кластеризация

Задача кластеризации

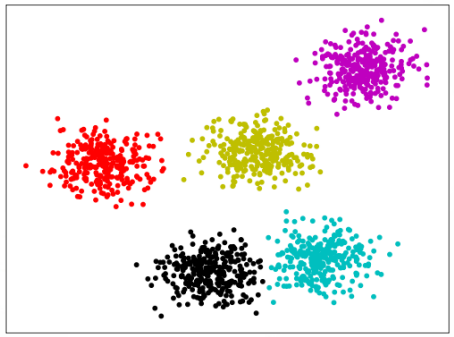
Работа в командах

Определяемся с требованиями

Метрики качества

K-means

Постановка задачи кластеризации



Дана выборка объектов $X = (x_1, ..., x_n)$.

Задача кластеризации:

выявить в данных K кластеров.

Кластер может быть:

- *Подвыборкой*
  Построить правило $f : X \to \{1, ..., K\}$,
  определяющее номер кластера только для объектов выборки.
- *Областью пространства*
  Построить правило $f : \mathscr{X} \to \{1, ..., K\}$,
  определяющее номер кластера для любых объектов пр-ва $\mathscr{X}$.
- *Нежестким*
  Построить правило $f(x) = (p_1, ..., p_K)$,
  определяющее распределение объекта по кластерам,
  где $p_k : \mathscr{X} \to [0, 1]$ — вероятность принадлежности x к класт. k.

Число K может быть известно заранее, т.е. гиперпараметр.

Цели кластеризации

- Упростить дальнейшую обработку данных.
  Разбить выборку X на группы схожих объектов
  и работать с каждой группой отдельно.

- Сократить объем хранимых данных.
  Например, оставить лишь по одному представителю
  из каждого кластера.

- Выделить нетипичные объекты.
  Объекты, которые не подходят ни к одному из кластеров.

- Использовать для разбиения данных на группы.
  Аналог классификации в случае, если нет целевых меток.
  Например, можно кластеризовать клиентов
  и разным группам предлагать разные услуги.

Пример: кластеризация пользователей

Цель: выделить кластеры схожих по поведению пользователей.

Данные:

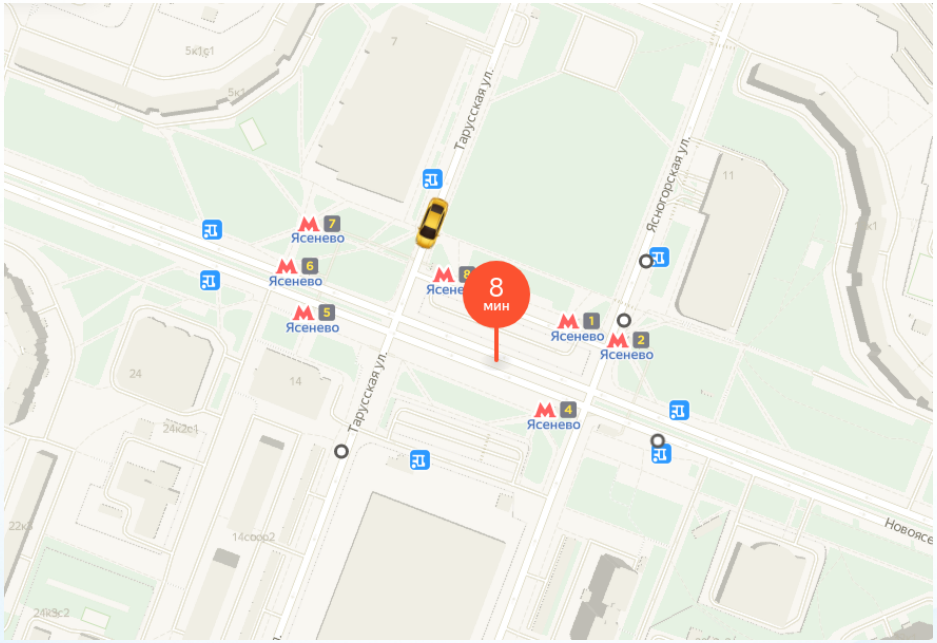
- Номер карты лояльности
- Фамилия Имя
- Пол
- Возраст
- Город, регион
- Номер телефона
- Вид деятельности
- Дата регистрации в программе лояльности
- Сумма покупок
- Частота посещений
- Средний чек
- Часто покупаемые категории продуктов

Полученные кластеры можно проинтерпретировать, проанализовав отличие по признакам. Например, *мужчины в возрасте от 30 до 40, посещают в среднем 5 раз в неделю, чаще всего покупают готовую еду.* Этот кластер скорее всего характеризует офисных работников, которые посещают супермаркет в обеденный перерыв.

Пример: точки посадки в такси

Цель:

определить наиболее удобные точки, где пассажир садится в такси.



Возможные проблемы:

- Неточность GPS-сигнала — в некоторых случаях погрешность может составлять 100 метров и более.
- Водитель может отметить начало поездки в приложении не сразу.

Работа в командах

Задача № 0. Кластеризоваться в команды по 4-5 человек.

Задача: время 5 минут

Имеется несколько тысяч супермаркетов.

Цель: кластеризовать магазины по похожести,
внутри кластеров выявить успешные магазины,
определить для остальных магазинов в кластере точки роста.

Данные:

- Продажи и выручка за каждый день, в т.ч. по отдельным товарам и категориям
- Потери от списаний
- Площадь торгового зала
- Количество сотрудников
- Торговые центры рядом
- Другие продуктовые магазины рядом
- Плотность и кол-во населения
- Городской трафик
- Образов. учреждения рядом

Вопросы: как кластеризовать, какие требования к модели,
как измерить качество?

Типы кластерных структур

Шарообразные

Ленточные

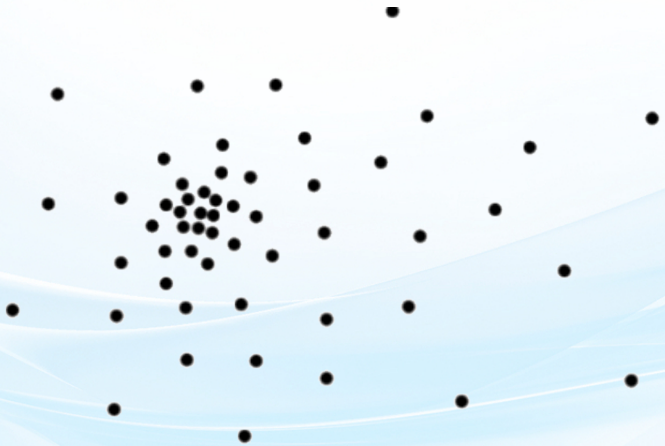
Соединяются перемычками

Разреженный фон
из шумовых объектов

Типы кластерных структур

Могут перекрываться

Сходство не по расстоянию



Кластеры отсутствуют

Неоднозначность задачи кластеризации

Решение задачи кластеризации неоднозначно:

- Точной постановки задачи кластеризации нет.
- Существует множество критериев качества.
- В реальных задачах метрикой качества кластеризации не редко оказывается моральная удовлетворенность заказчика
- Число кластеров K обычно не известно заранее.
- Результат зависит от выбора метрики расстояния (схожести) между объектами.

Пример: Сколько здесь кластеров?

Задача: время еще 5 минут

Имеется несколько тысяч супермаркетов.

Цель: кластеризовать магазины по похожести,
внутри кластеров выявить успешные магазины,
определить для остальных магазинов в кластере точки роста.

Данные:

- Продажи и выручка за каждый день, в т.ч. по отдельным товарам и категориям
- Потери от списаний
- Площадь торгового зала
- Количество сотрудников
- Торговые центры рядом
- Другие продуктовые магазины рядом
- Плотность и кол-во населения
- Городской трафик
- Образов. учреждения рядом

Вопросы: как кластеризовать, как измерить качество?

Кластеризация

Задача кластеризации

Работа в командах

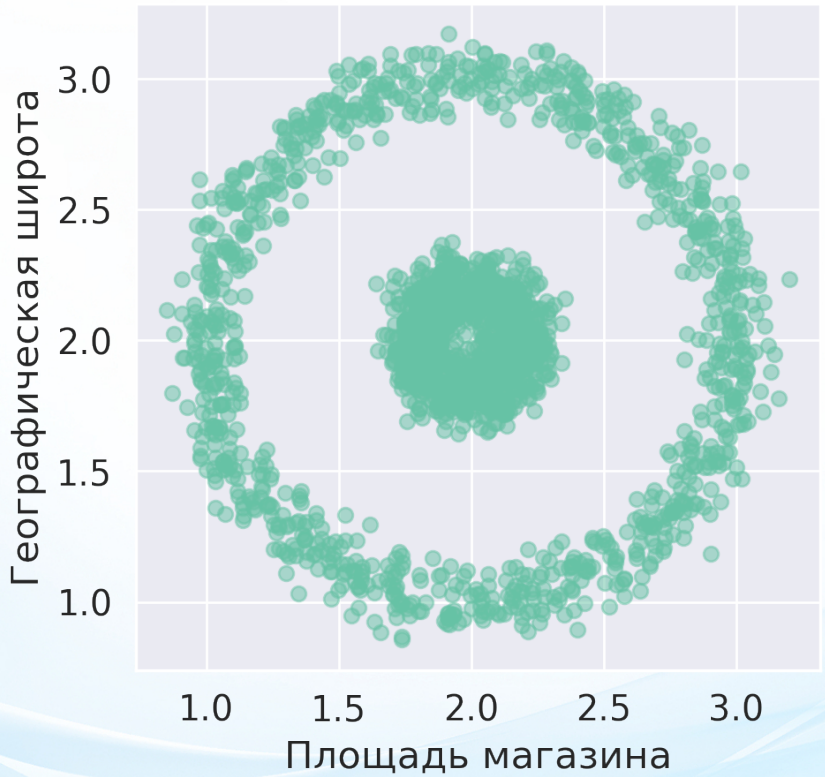
Определяемся с требованиями

Метрики качества

K-means

Какие требования к форме кластеров?

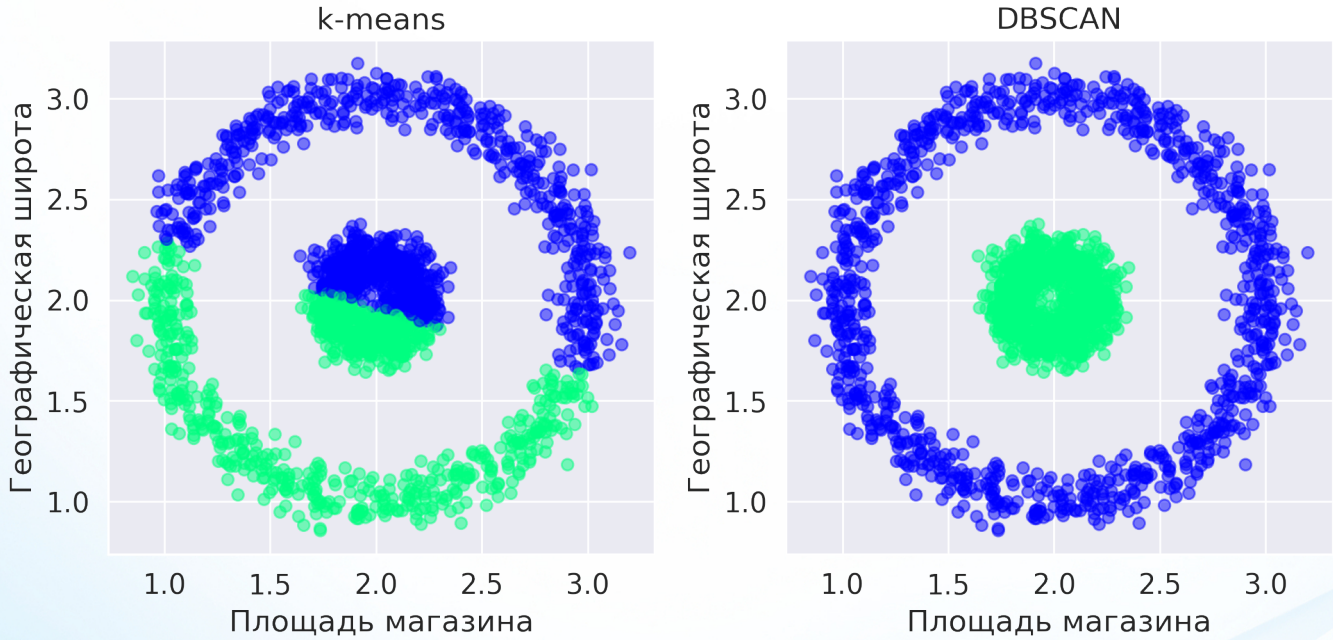
Пусть данные выглядят так:



Сколько здесь кластеров? Какие они?

Какие требования к форме кластеров?

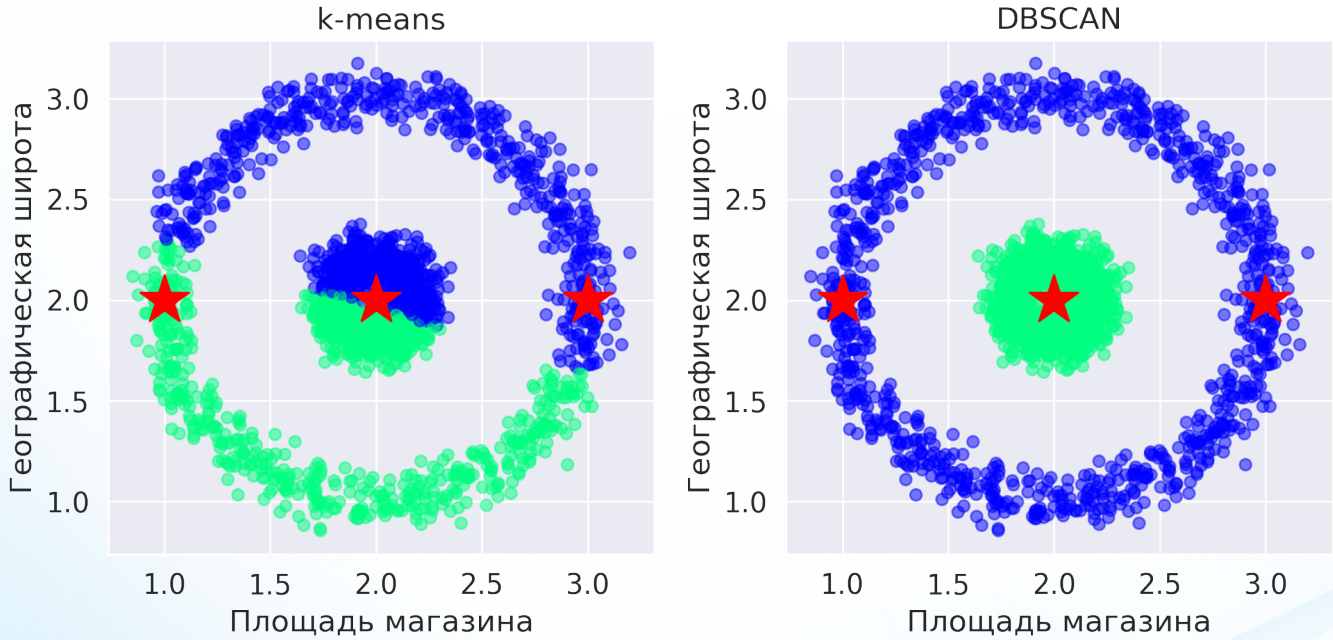
Применим два популярных метода для кластеризации на 2 кластера:



Какой лучше и почему?

Какие требования к форме кластеров?

Применим два популярных метода для кластеризации на 2 кластера:



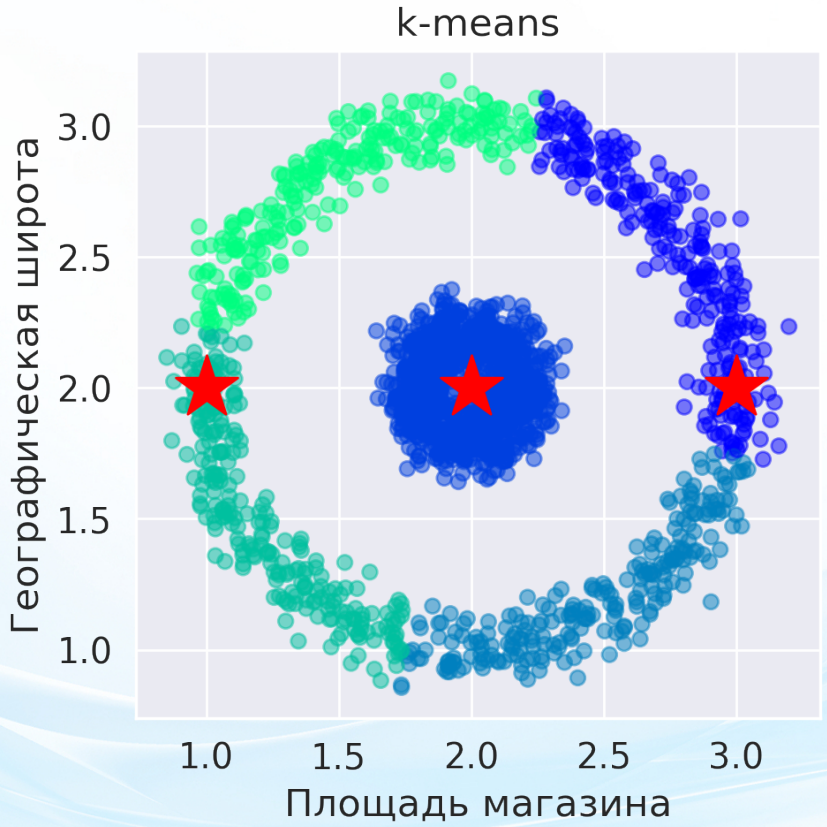
Какой лучше и почему?

Для ответа на этот вопрос рассмотрим три магазина разной площади на одной широте.
Видим, что второй метод дает неинтерпретируемый результат: магазины с площадью 1 и 3 лежат в одном кластере, а магазин с площадью 2 — в другом.

Какие требования к форме кластеров?

Достаточное ли количество кластеров?

Возьмем больше для первого метода:



Кажется, так лучше.

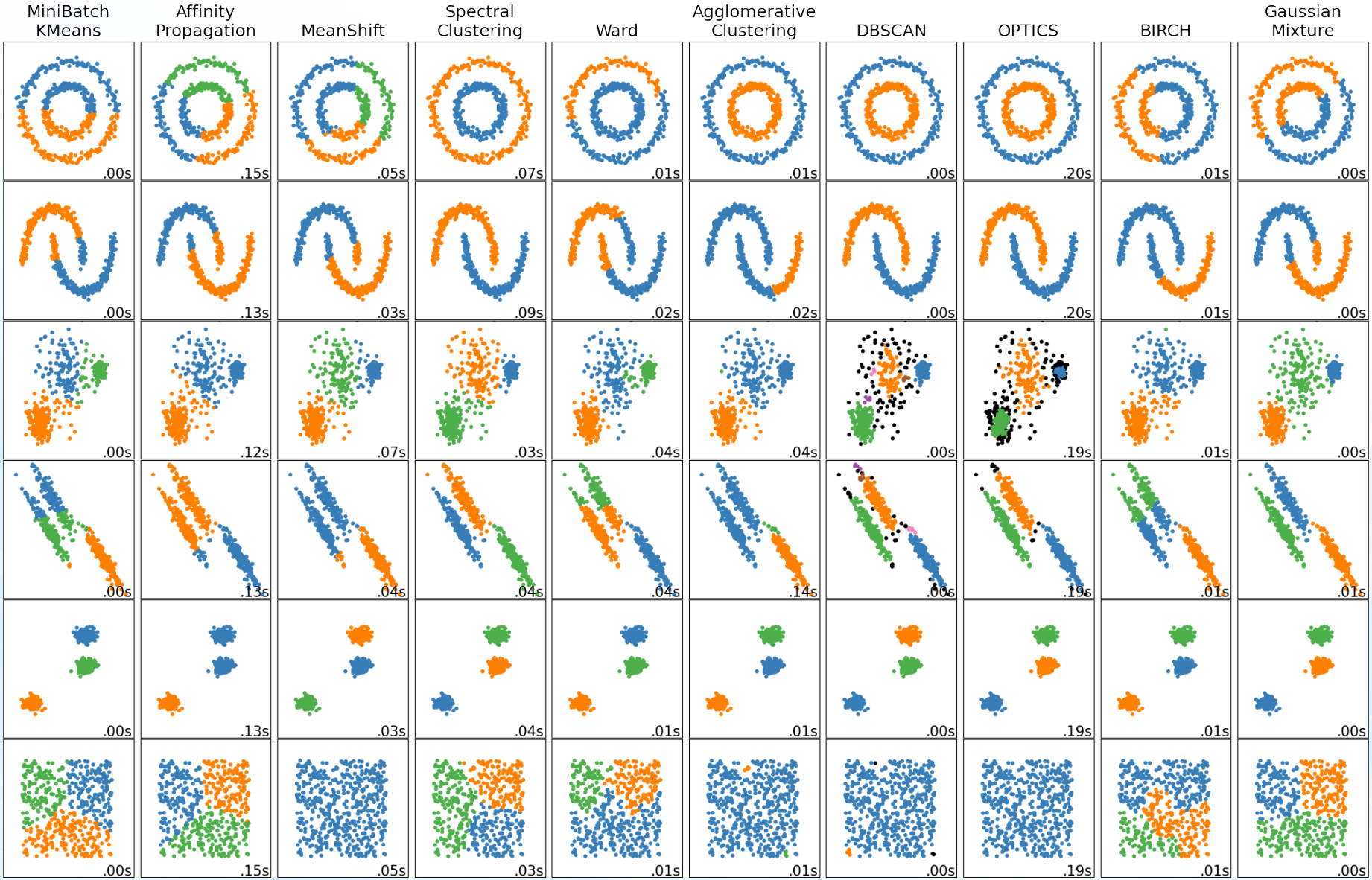
Какие требования к форме кластеров?

Итог:

- Нужно понимать, какую форму кластера можно получить в зависимости от применяемого метода.
- Для решаемой задачи нужно понять, каким свойством должен обладать кластер.
- Если нужны интерпретируемые для бизнеса кластеры, то они обязательно должны быть *выпуклыми*.

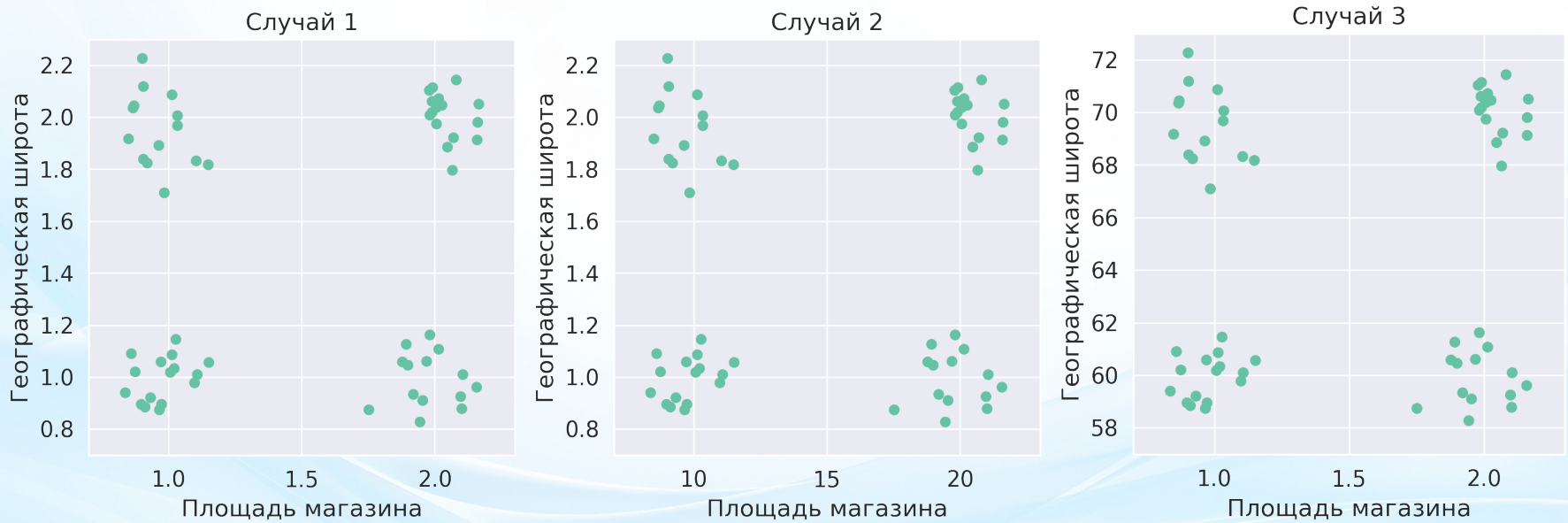
  Желательно также получать более компактные кластеры, нежели сильно растянутые.
- Если же кластеризация используется какой-либо ML-моделью, то форма кластера может быть не сильно важна, стоит следить за качеством ML-модели.

Сравнение разных методов



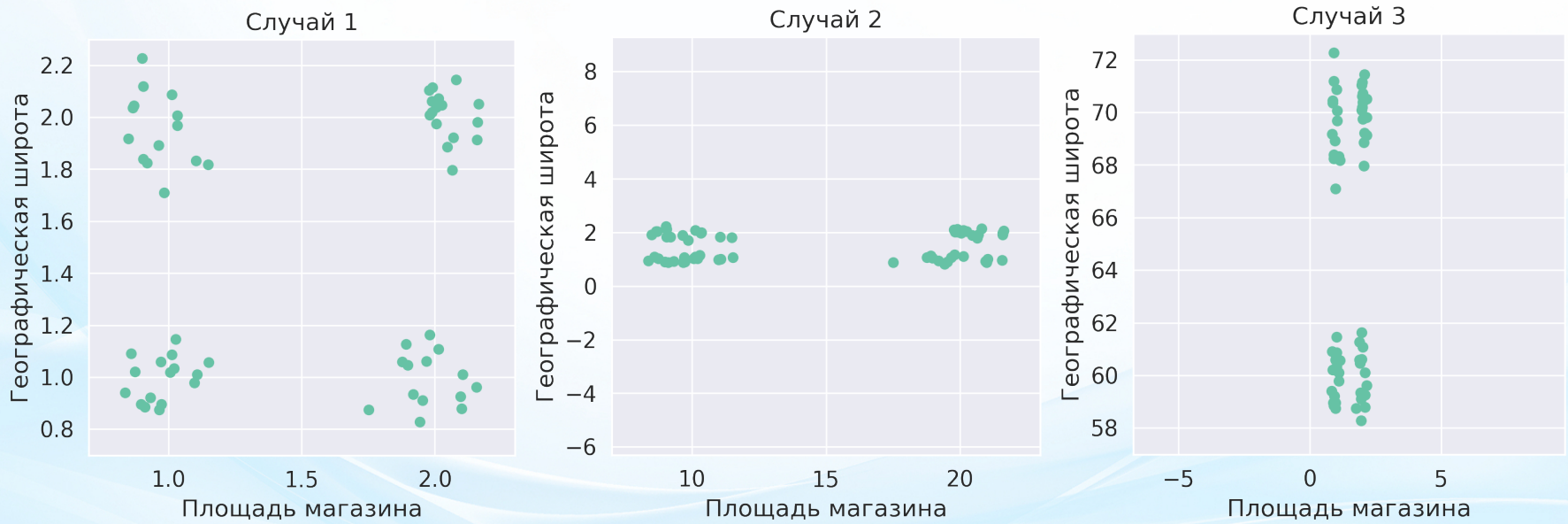
Подумаем еще

Посмотрим на три набора данных. В чем их отличие?



Подумаем еще

Как на них посмотрит метод кластеризации?



Масштабирования и расстояния

Итог:

- Результат кластеризации сильно зависит от используемой метрики (функции расстояния).
- Если предполагается ручной анализ кластеров, то не стоит выбирать неинтерпретируемые метрики и выполнять неинтерпретируемые преобразования признаков.
- В простом случае стоит выполнять стандартизацию признаков. Иначе результат кластеризации во многом будет определяться признаком, который имеет самый большой диапазон значений.
- В идеале стоит подумать, какие признаки более значимы. Например, потребовать, чтобы изменения были сопоставимыми:
  1. Площадь магазина увеличилась на 10 m^2 при той же широте
  2. Широта магазина увеличилась на x при неизменной площади

  Исходя из желаемого значения x расставить веса признакам после стандартизации.

Кластеризация

Задача кластеризации

Работа в командах

Определяемся с требованиями

Метрики качества

K-means

Расстояния внутри и между кластерами

Пусть задана функция расстояния между объектами — $\rho(x_1, x_2)$ и f — построенный метод кластеризации.

1. Среднее внутрикластерное расстояние:

$$F_0(f) = \frac{\sum\limits_{i<\ell} I\{f(x_i) = f(x_\ell)\} \cdot \rho(x_i, x_\ell)}{\sum\limits_{i<\ell} I\{f(x_i) = f(x_\ell)\}} \longrightarrow \min_f$$

2. Среднее межкластерное расстояние:

$$F_1(f) = \frac{\sum\limits_{i<\ell} I\{f(x_i) \neq f(x_\ell)\} \cdot \rho(x_i, x_\ell)}{\sum\limits_{i<\ell} I\{f(x_i) \neq f(x_\ell)\}} \longrightarrow \max_f$$

3. $F_0(f)/F_1(f) \longrightarrow \min\limits_f$

Метрика 1 не подходит для выбора количества кластеров:
её оптимум достигается, если все кластеры — одноэлементные.

Расстояния внутри и между кластерами

4. Среднее расстояние до центра кластера.

   Будем считать, что каждый кластер характеризуется своим *центром* μ_k.

$$\sum_{k=1}^{K} \sum_{i=1}^{n} I\{f(x_i) = k\} \cdot \rho(x_i, \mu_k)$$

5. Индекс Данна (Dunn Index):

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

   $d(k', k)$ — расстояние между кластерами k' и k.
   Например, евклидово расстояние между центрами кластеров.

   $d(k)$ — внутрикластерное расстояние для кластера k.
   Например, сумма расстояний от всех объектов кластера k до его центра.

Силуэт

Пусть точка x лежит в кластере C_k.

a_x — среднее расст. от x до всех других объектов из его же кластера:

$$a_x = \frac{1}{|C_k| - 1} \sum_{z \in C_k, z \neq x} \rho(x, z)$$

b_x — среднее расстояние. от x до всех объектов
из ближайшего другого кластера:

$$b_x = \min_{\ell \neq k} \frac{1}{|C_\ell|} \sum_{z \in C_\ell} \rho(x, z)$$

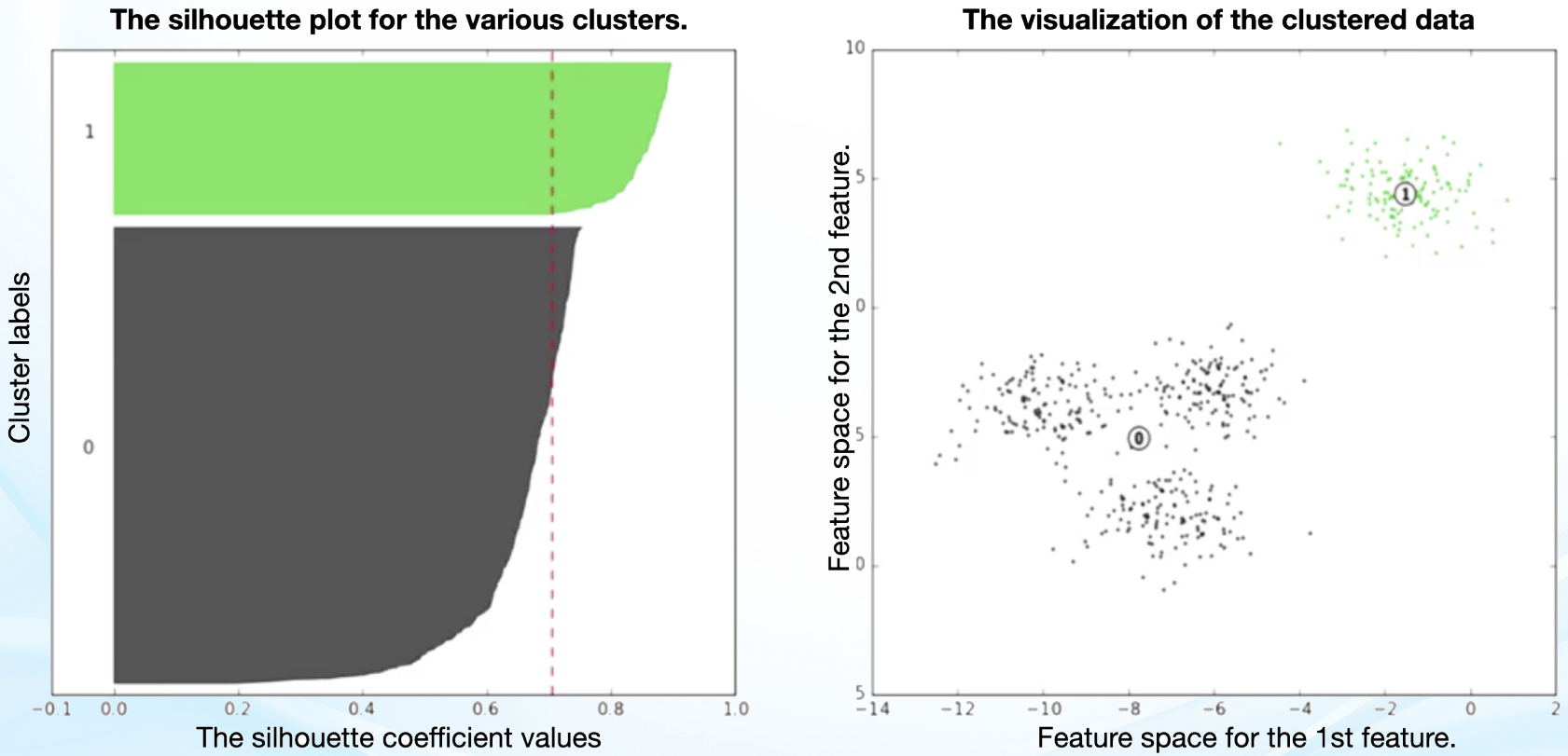
Силуэт для точки x: $$s_x = \frac{b_x - a_x}{\max(b_x, a_x)}$$

Если $b_x \gg a_x$ — хороший случай, то s_x около 1.

Если $b_x \ll a_x$ — плохой случай, то s_x около -1.

Средний коэффициент силуэта по выборке: $s = \frac{1}{n} \sum_{i=1}^{n} s_{x_i}$

Силуэт

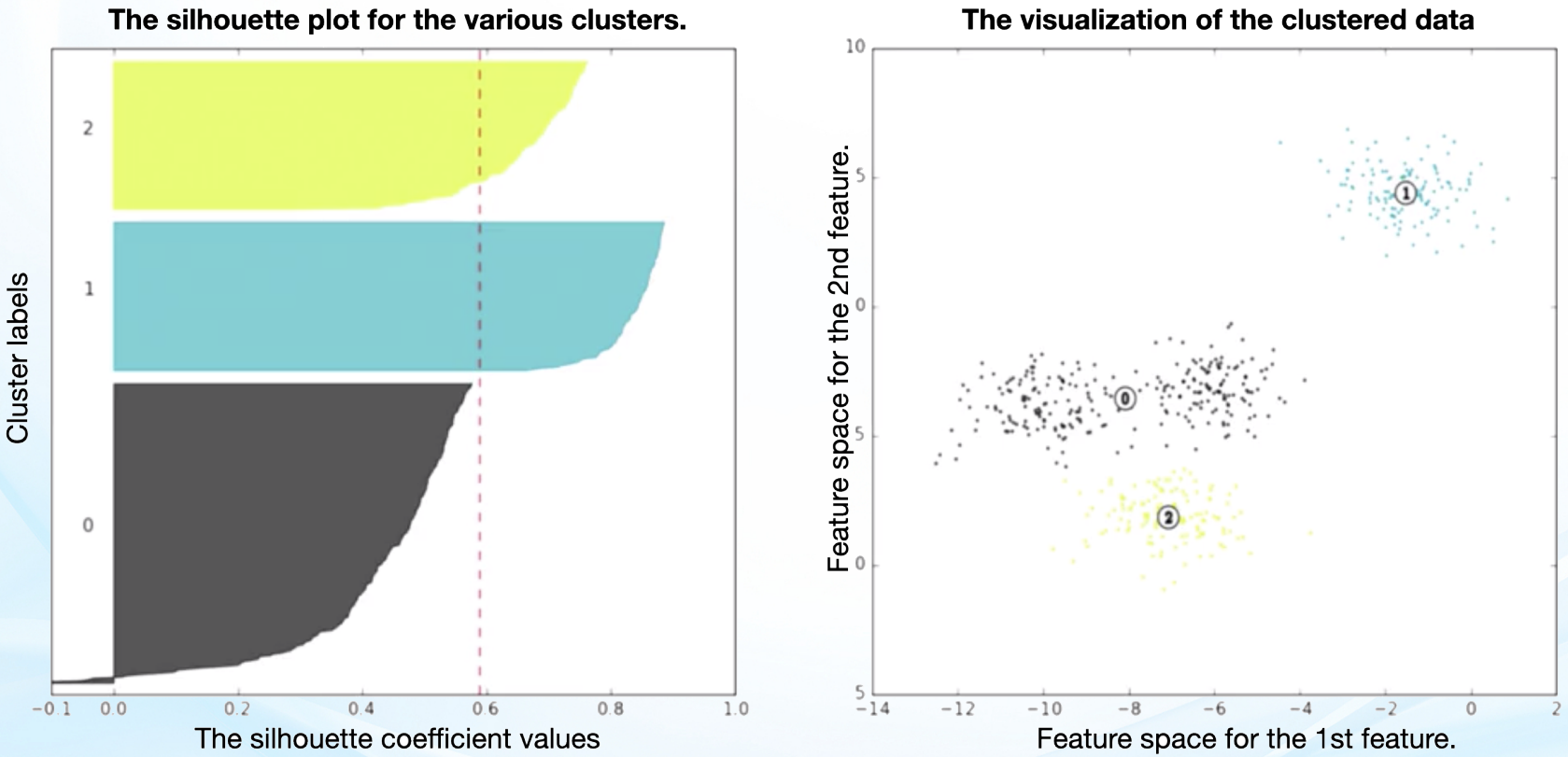


Пунктирная линия - среднее значение силуэта по выборке.

Разброс значений силуэта между кластерами не очень большой.

Кластеризация считается хорошей.

Силуэт

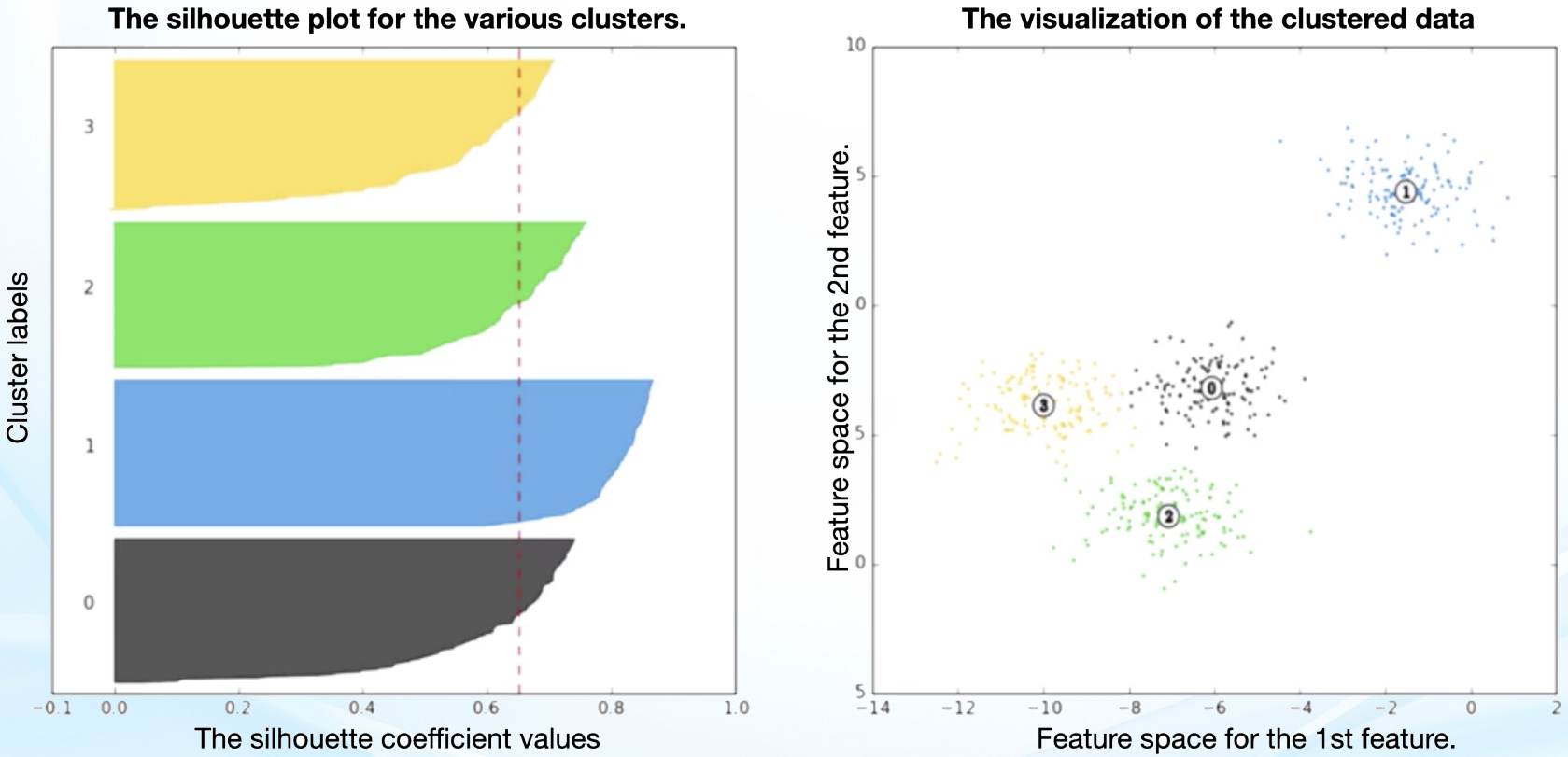


Разброс значений силуэта между кластерами большой.

Значения силуэта для кластера 0 ниже среднего значения.

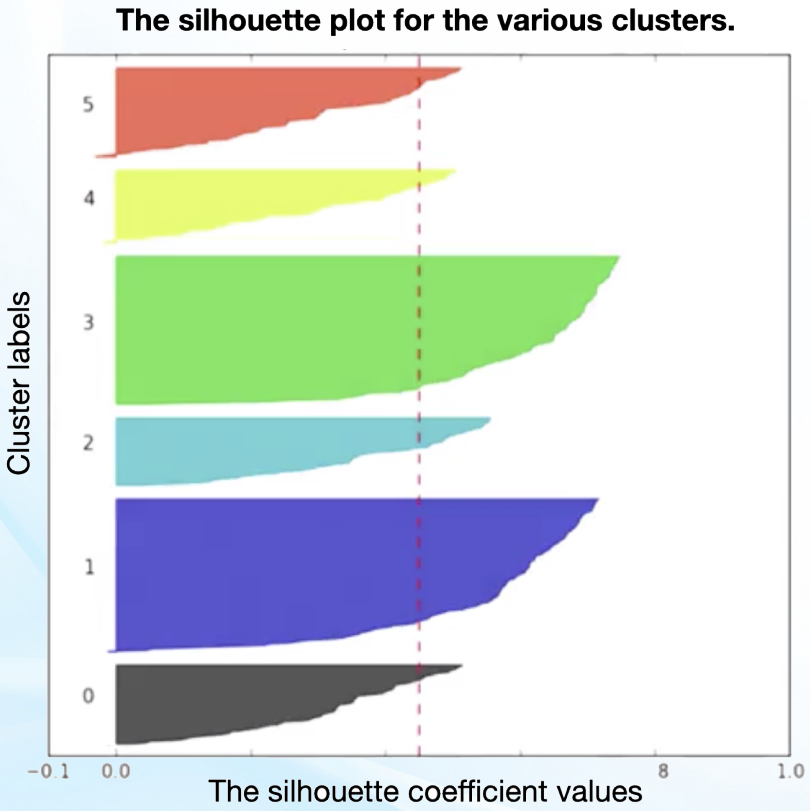
Кластеризация считается плохой.

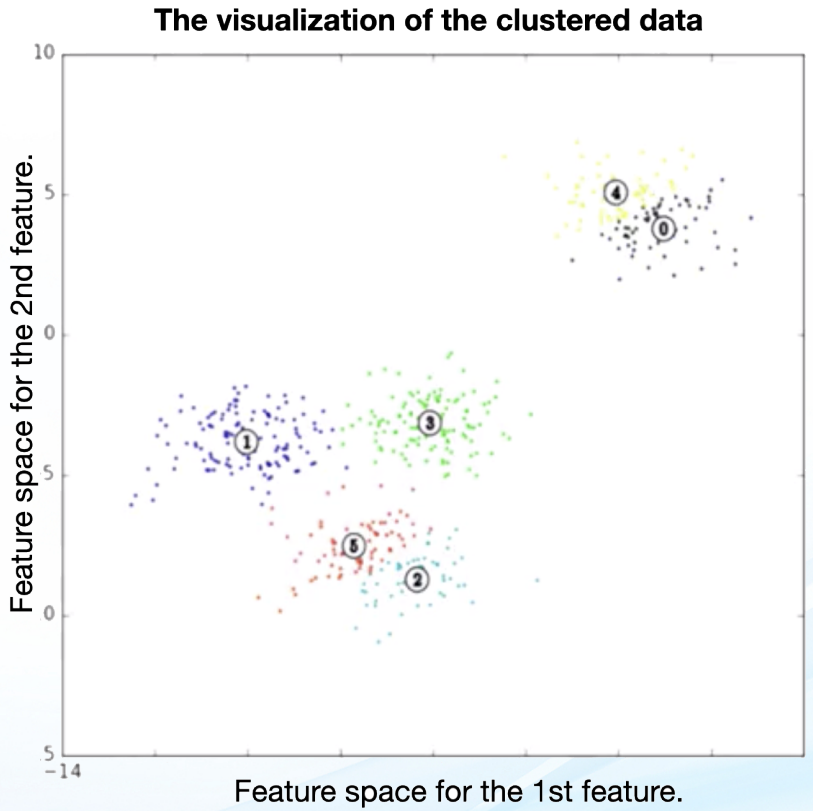
Силуэт



Разброс значений силуэта между кластерами не очень большой.
Кластеризация считается хорошей.

Силуэт

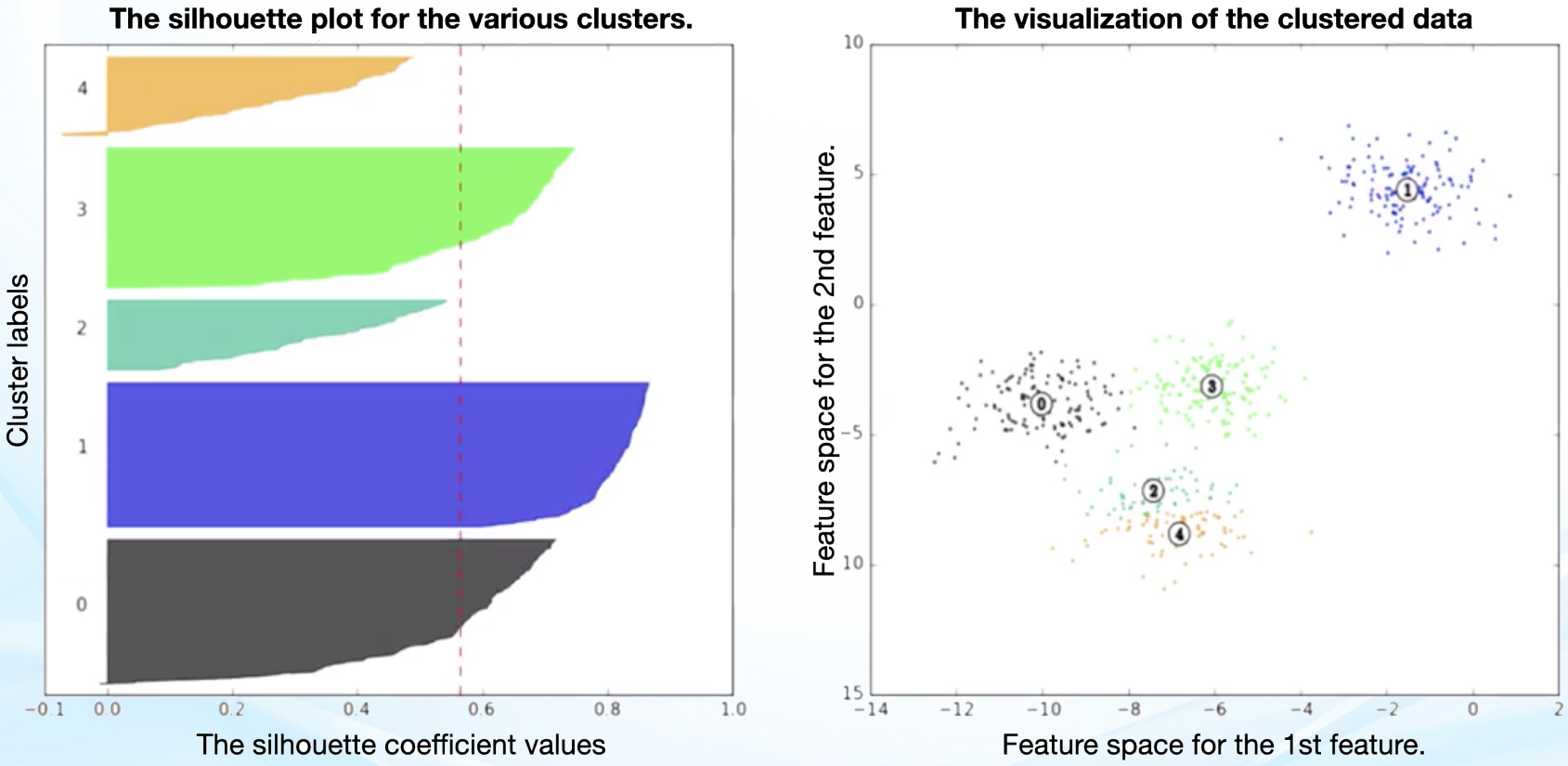




Разброс значений силуэта между кластерами большой.

Кластеризация считается плохой.

Силуэт



Разброс значений силуэта между кластерами большой.

Значения силуэта для кластеров 2 и 4 ниже среднего значения.

Кластеризация считается плохой.

Как поступать на практике

Часто на практике:

- "Внутренние" метрики не интерпретируемы и не согласуются с желаниями заказчика.
- Объекты плохо разбиваются на кластеры или кластеров нет вообще. А заказчик хочет кластеры.

- Необходимо просто и наглядно убедить заказчика, что результат кластеризации хороший.

Как поступать на практике

- Заказчика может устроить подробная интерпретация каждого кластера по совокупности признаков.
- Возможно, есть разметка объектов, которые должны лежать в одном или разных кластерах. В таком случае стоит посчитать на них метрики качества классификации.
- Обычно кластеризация — промежуточная задача для решения другой задачи. В таком случае можно использовать такой критерий качества, который согласуется с качеством целевой задачи.
  - Если цель — на кластерах построить разные регрессионные модели, то можно оценивать по MSE весь пайплайн предобработка -> кластеризация -> регрессия.
  - Если цель — выявить успешные магазины в каждом кластере, то можно максимизировать среднюю внутрикластерную дисперсию выручки среди "хороших" кластеризаций.

Кластеризация

Задача кластеризации

Работа в командах

Определяемся с требованиями

Метрики качества

K-means

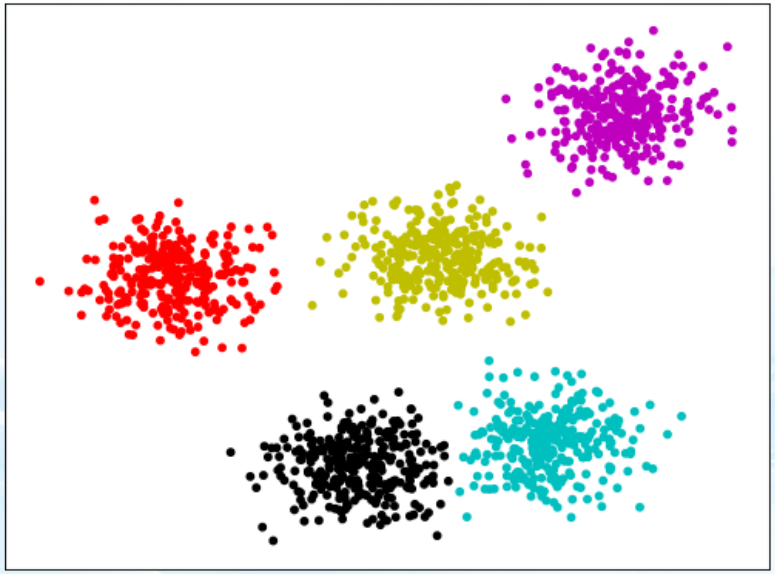
K-means (метод K-средних)

Дана выборка $X = (x_1, ..., x_n)$.

Задано число кластеров K.

Нужно построить отображение $f : \mathscr{X} \longrightarrow \{1, ..., K\}$,

то есть отнести каждый объект к одному из кластеров.



K-means (метод K-средних)

В качестве метрики расстояния между объектами обычно используется евклидова метрика: $\rho(x, z) = \|x - z\|^2$

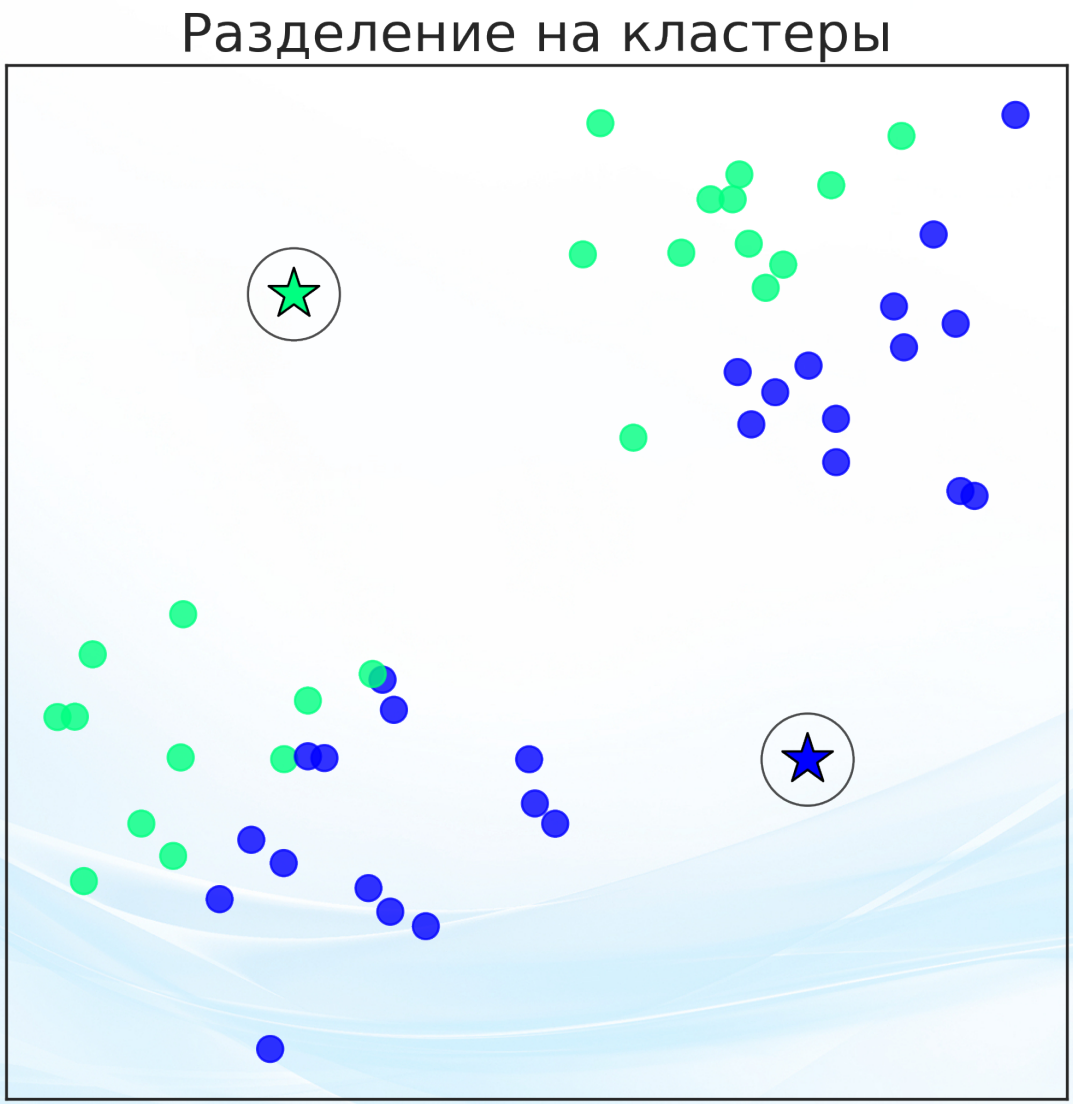
Процедура:

1. Задать начальное приближение центров кластеров $\mu_1, \mu_2, .., \mu_K$.
2. Повторять
   - 2.1 Отнести каждый объект к ближайшему центру:
   $$f(x_i) = \arg\min_k \|x_i - \mu_k\|^2.$$
   - 2.2 Вычислить новые положения центров:
   $$\mu_k = \frac{\sum_{i=1}^{n} I\{f(x_i) = k\}\, x_i}{\sum_{i=1}^{n} I\{f(x_i) = k\}}$$
   - 2.3 Пока $f(x_i)$ не перестанут изменяться.

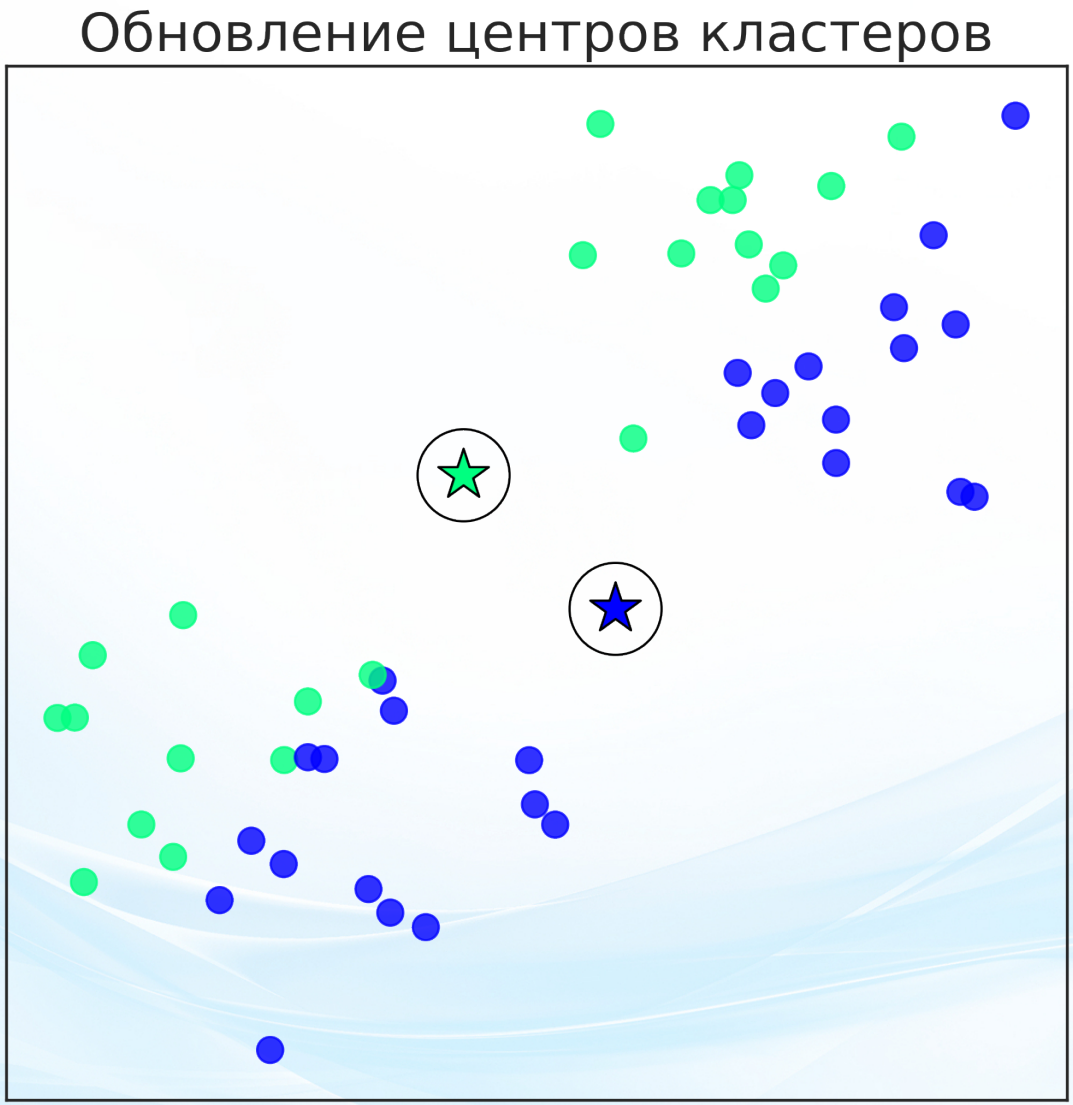
Метод применим и к новым данным: берем ближайший к точке кластер.

K-means: пример работы

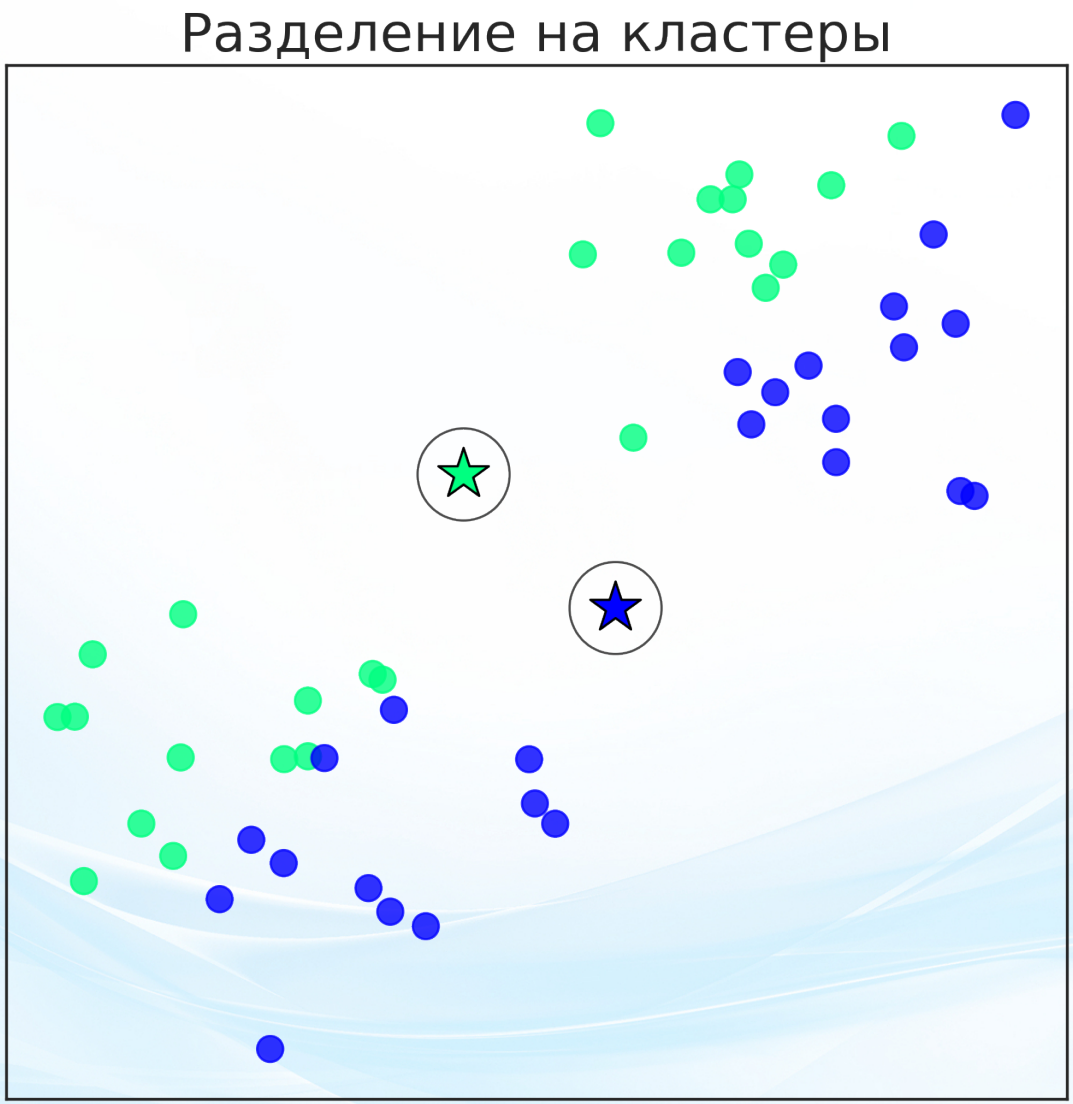
K-means: пример работы



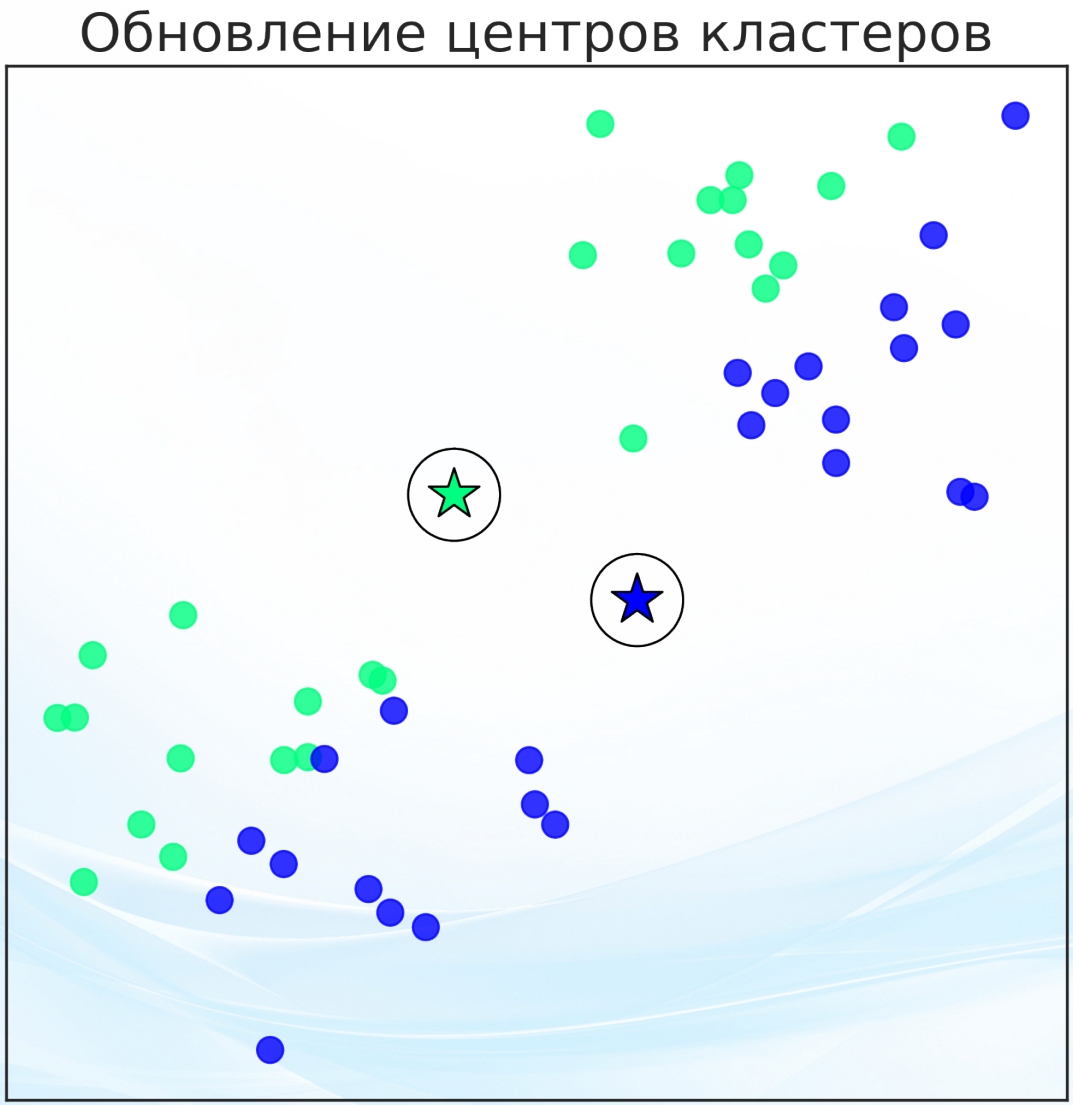
K-means: пример работы



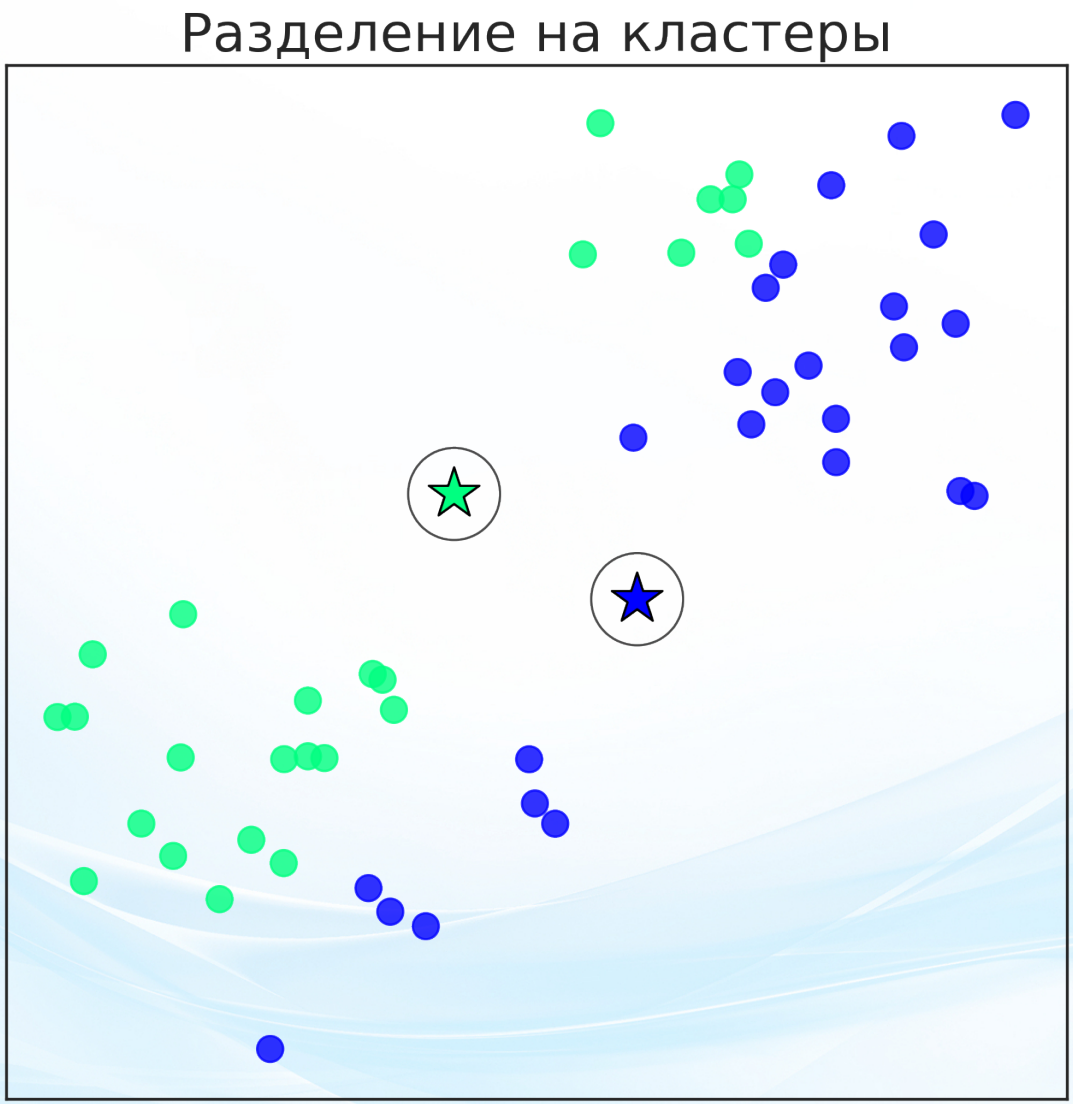
K-means: пример работы



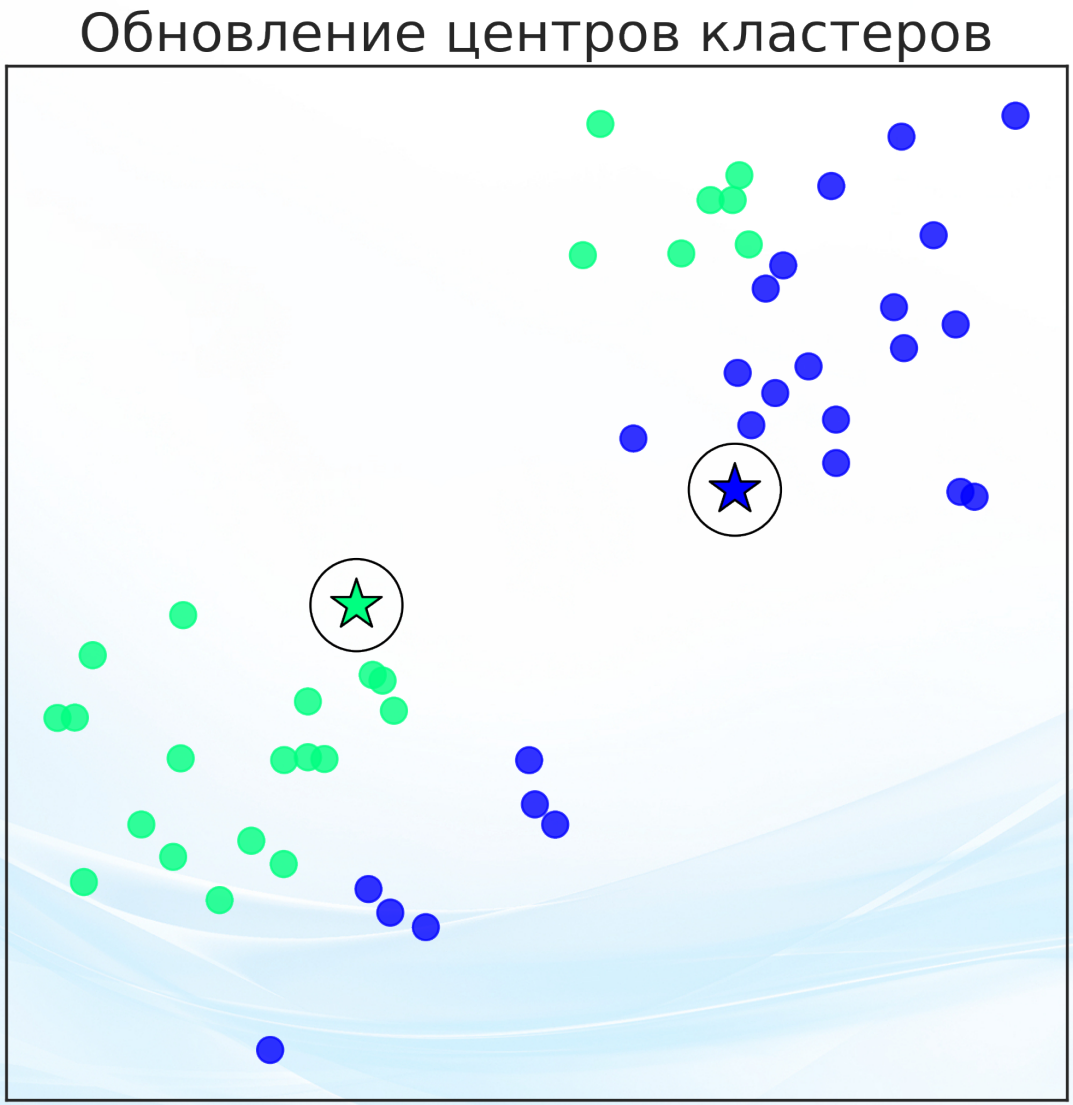
K-means: пример работы



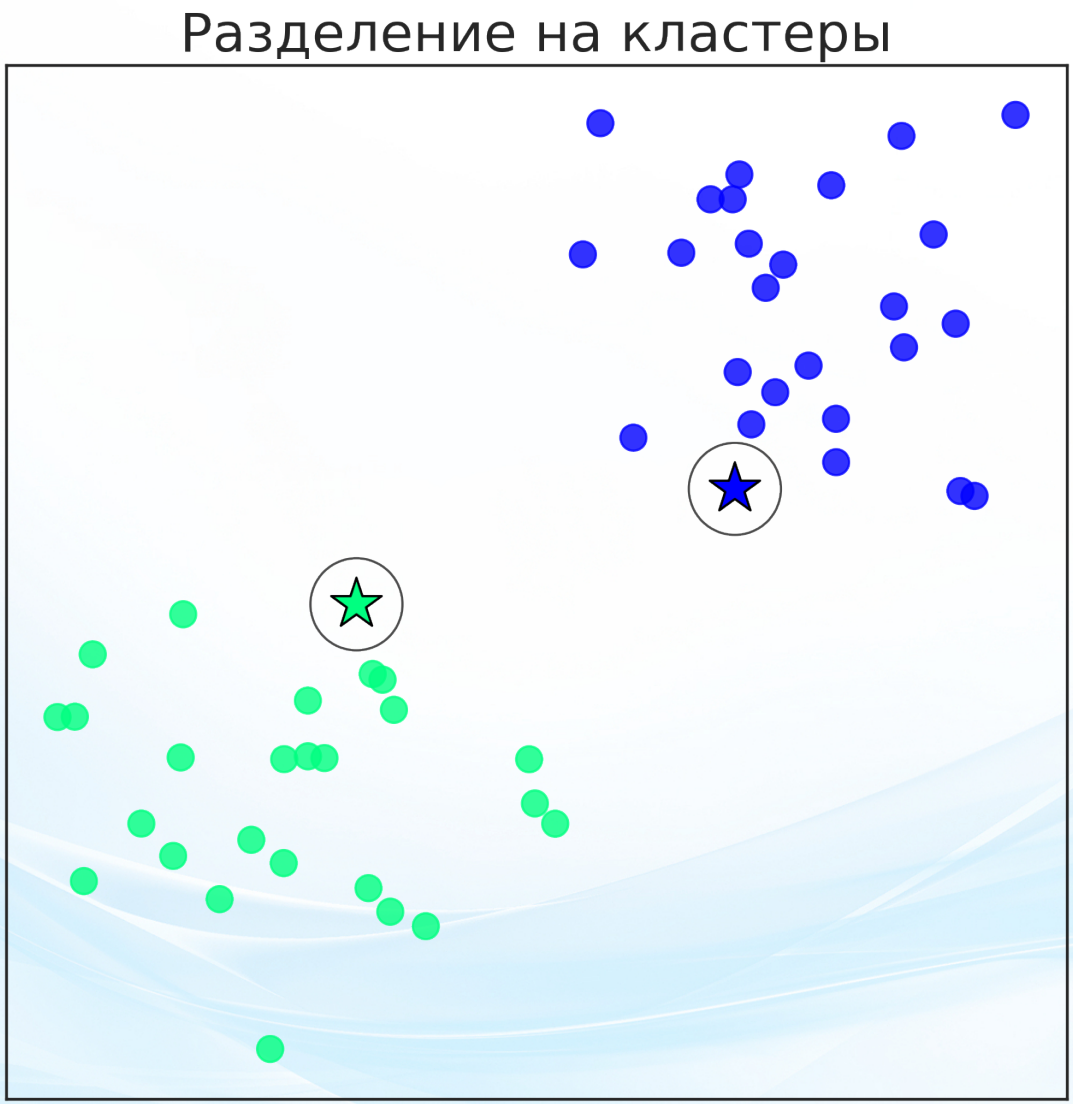
K-means: пример работы



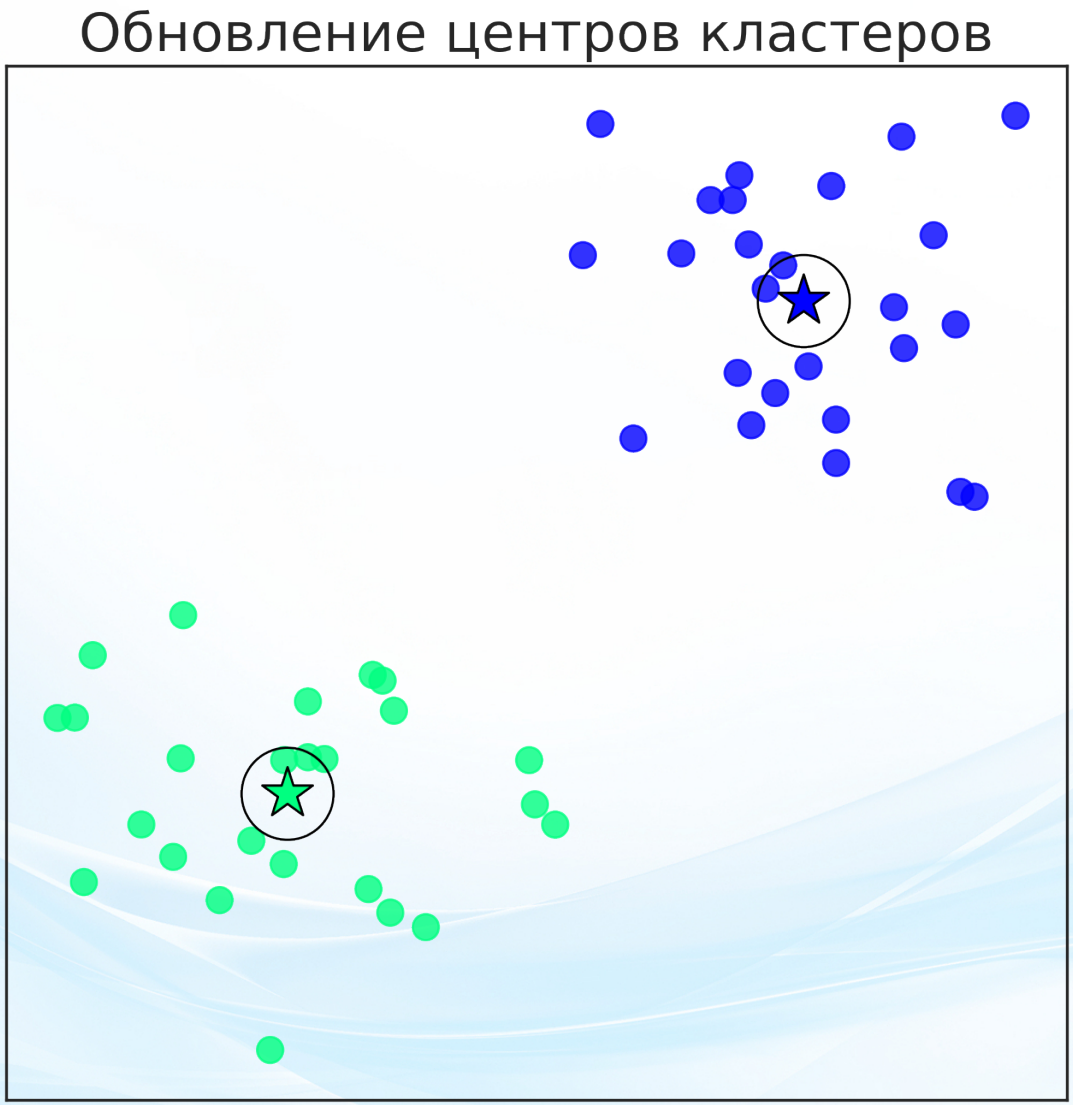
K-means: пример работы



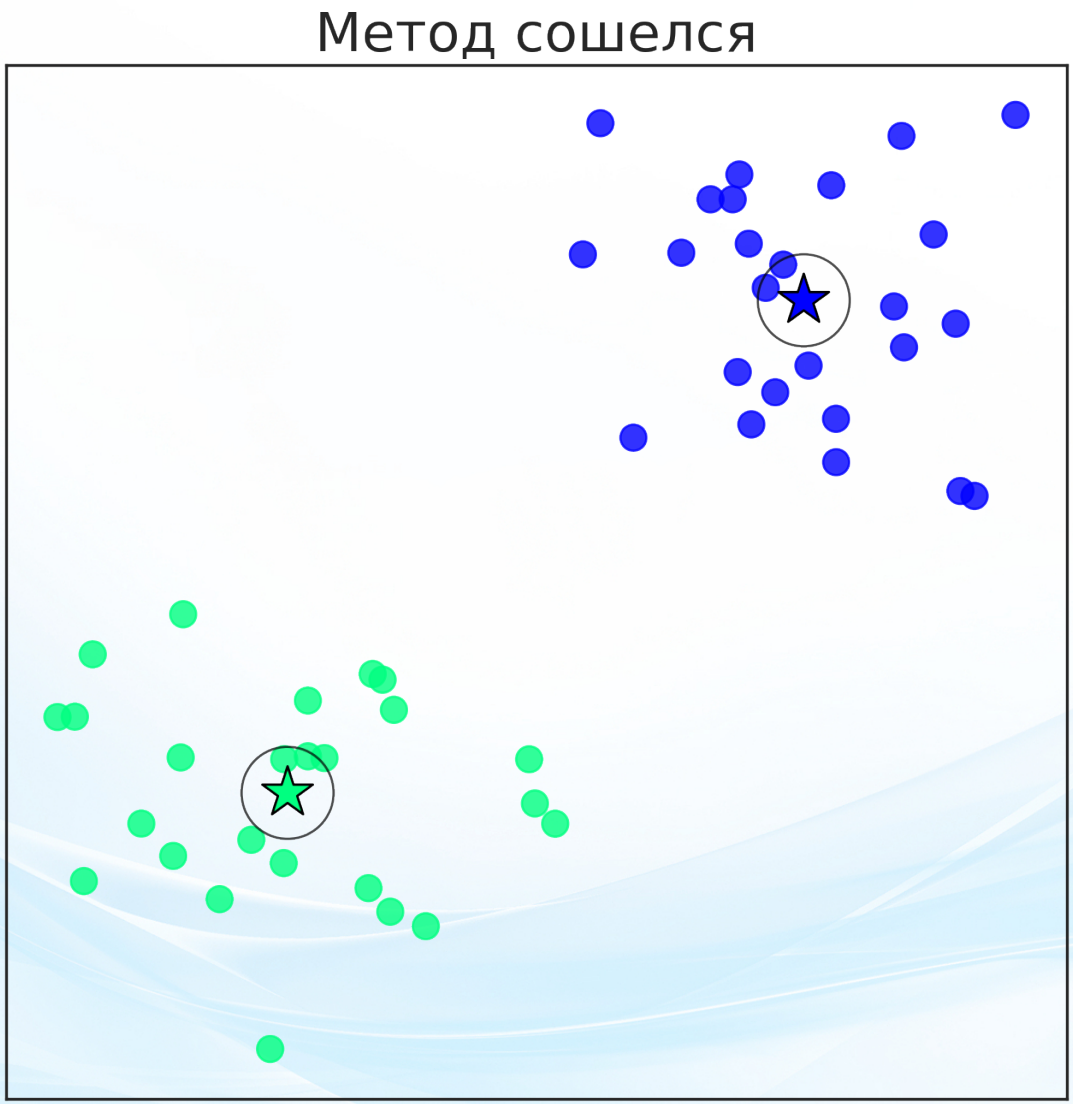
K-means: пример работы



K-means: пример работы



K-means: пример работы



Что оптимизирует K-means

Утверждение.
K-means оптимизирует сумму квадратов внутрикластерных расстояний до центра кластера:

$$Q(f) = \sum_{i=1}^{n} \left\| x_i - \mu_{f(x_i)} \right\|^2 \to \min_{f,\mu}$$

Доказательство

Покажем, что на каждом шаге $Q(f)$ убывает или не изменяется:

- **Шаг 1:** *Отнести каждый объект к ближайшему центру.*
  Центры — фиксированные точки. При отнесении точки x_i к ближайшему центру не возрастает значение $\left\| x_i - \mu_{f(x_i)} \right\|^2$.
  $\Longrightarrow$ функционал $Q(f)$ не возрастает.

- **Шаг 2:** *Вычислить новые положения центров.*
  Кластеры фиксированы. Функционал $Q(f)$ разбивается на K независимых слагаемых, для минимизации которых в качестве центра нужно взять среднее: $\frac{1}{|i:\ f(x_i)=k|} \sum\limits_{i:\ f(x_i)=k} x_i$.
  $\Longrightarrow$ функционал $Q(f)$ не возрастает.

Что оптимизирует K-means

Утверждение.
K-means оптимизирует сумму квадратов внутрикластерных расстояний до центра кластера:

$$Q(f) = \sum_{i=1}^{n} \left\| x_i - \mu_{f(x_i)} \right\|^2 \to \min_{f,\mu}$$

Доказательство

Доказали, что на каждой итерации $Q(f)$ не возрастает.

Равенство $Q(f)$ между итерациями достигается
либо если отображение объектов в кластеры не меняется,
либо в редких случаях симметрии.

Метод найдет лишь локальный минимум функционала $Q(f)$.
Нахождение глобального минимума — NP-полная задача.

Особенности

1. Сходится к локальному оптимуму,
   имеет смысл запускать из разных начальных приближений.

2. Кластеры представляют собой выпуклые множества.

3. **Mini-batch k-means**

   На каждом шаге:

   - Выбираем случайное подмножество объектов.
   - Распределяем это подмножество объектов по кластерам.
   - Считаем центры кластеров по данным объектам.

4. K-means — метрический метод, имеет смысл делать снижение размерности: отбор признаков, PCA, UMAP, t-SNE и прочее.

K-means++

В зависимости от начального приближения центров кластеров

- может потребоваться разное время для сходимости;
- результаты могут получиться разными.

Решение: брать центры подальше друг от друга.

Как?

1. Первый центр выбираем случайно из равномерного распределения на выборке.
2. Каждый следующий центр выбираем случайно по некоторой вероятности из оставшихся точек.

   При этом вероятность выбрать каждую точку пропорциональна квадрату расстояния от нее до ближайшего выбранного центра.

EM-алгоритм (для справки)

Нежесткая кластеризация

Пусть объект x не строго принадлежит одному кластеру,
а имеет некую вероятность принадлежности к каждому кластеру.

Предположим, что имеется K кластеров.

Причем объекты подчиняются модели смеси распределений
с плотностью:

$$p(x) = \sum_{k=1}^{K} \pi_k p_{\theta_k}(x), \qquad \pi_k \geqslant 0, \quad \sum_{k=1}^{K} \pi_k = 1$$

где π_k — вероятность получить объект из кластера k,

$p_{\theta_k}(x)$ — плотность объекта внутри кластера k.

Смысл параметров:

- π_k отвечают за соотношение кластеров,
- θ_k за положения кластеров в пространстве.

EM-алгоритм (для справки)

Рассмотрим смесь гауссовских распределений.

$$p_{\theta_k} = \mathcal{N}(\mu_k, \Sigma_k)$$

E-шаг: Оцениваем вероятности принадлежности объектов к кластерам, при фиксированных параметрах распределений.

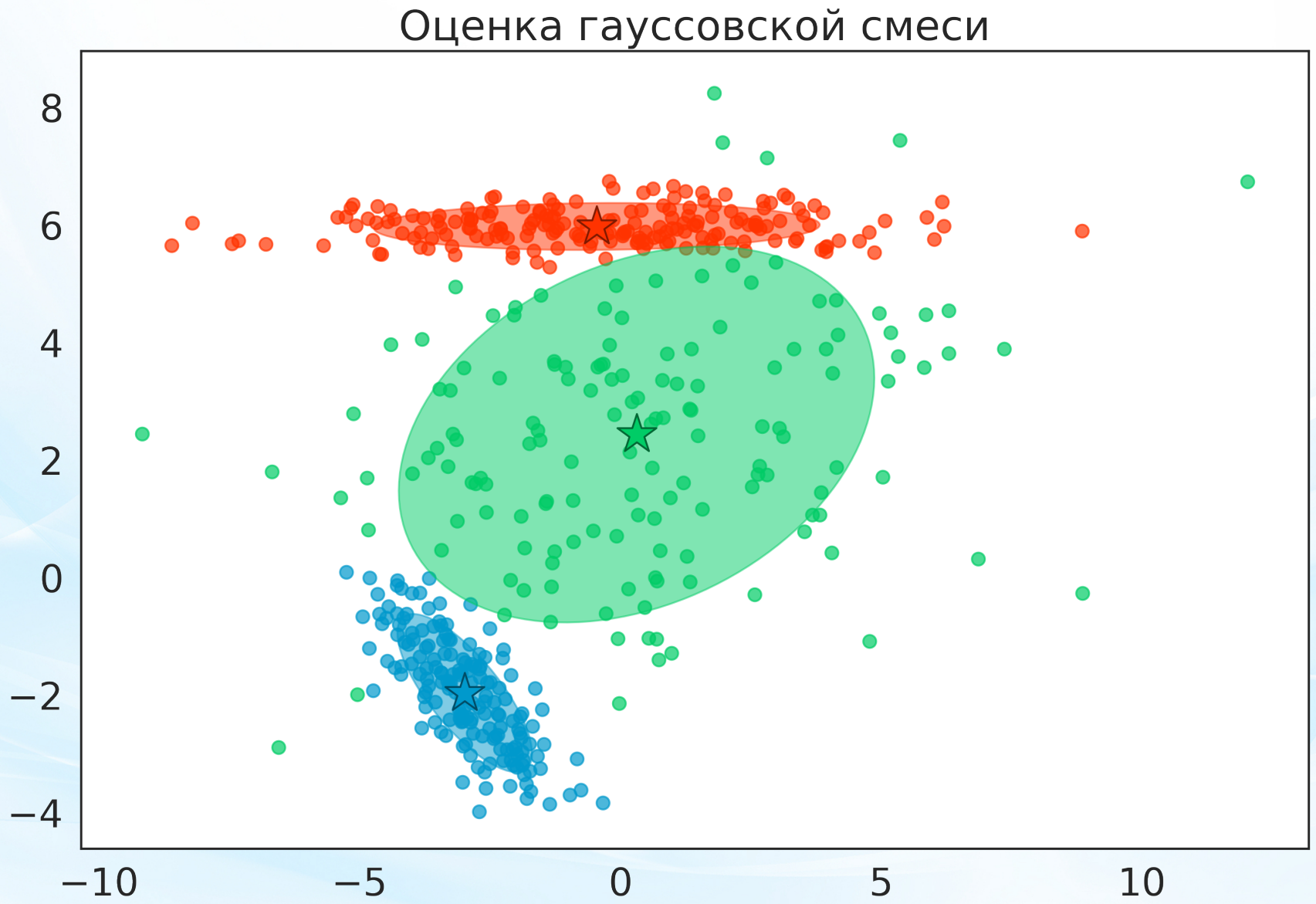
$$\gamma_{ik} = \mathrm{P}(x_i \text{ в кластере } k) = \frac{\pi_k \cdot p_{\mu_k \Sigma_k}(x_i)}{\sum\limits_{k=1}^{K} \pi_k \cdot p_{\mu_k \Sigma_k}(x_i)}$$

M-шаг: Оцениваем параметры распределений при фиксированных вероятностях принадлежности к кластерам.

$$\mu_k = \frac{\sum_{i=1}^{n} \gamma_{ik}\, x_i}{\sum_{i=1}^{n} \gamma_{ik}} \qquad \Sigma_j = \frac{\sum_{i=1}^{n} \gamma_{ik}(x_i - \mu_k)(x_i - \mu_k)^T}{\sum_{i=1}^{n} \gamma_{ik}} \qquad \pi_k = \frac{1}{n} \sum_{i=1}^{n} \gamma_{ik}$$

Повторяем до сходимости

EM-алгоритм: Пример



Сравним K-means и EM

EM

E-шаг:

Для каждого объекта оцениваем вероятность принадлежности к кластерам.

M-шаг:

Оцениваем параметры, задающие распределения.

K-means

E-шаг:

Для каждого объекта находим ближайший кластер.

M-шаг:

Оцениваем центры кластеров.

$\Rightarrow$ K-means — упрощенный вариант EM алгоритма для разделения смеси гауссовских распределений.

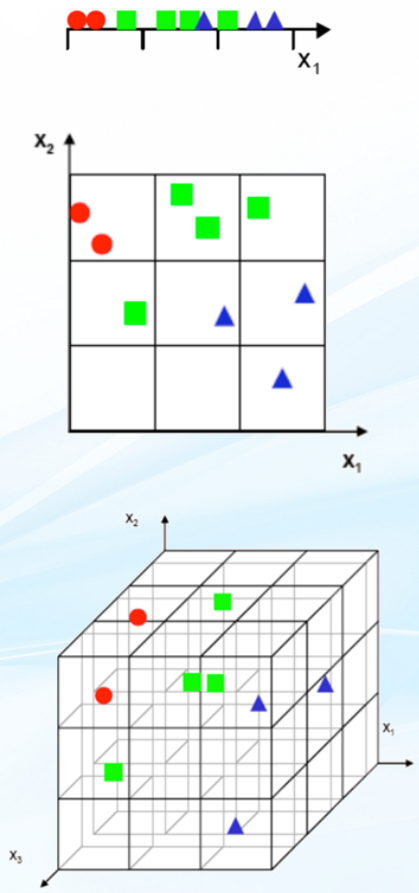
Понижение размерности

Проклятие размерности

Проклятие размерности — ситуация экспоненциального возрастания количества данных из-за увеличения размерности пространства.

Следствия:

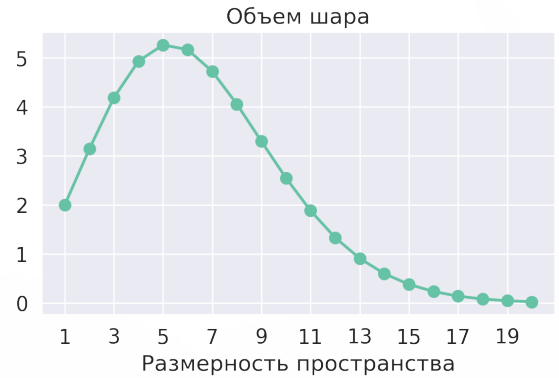
- Трудоемкость вычислений
- Хранение огромного количества данных
- Увеличение доли шумов
- В линейных моделях увеличение числа признаков ведет к проблемам мультиколлинеарности и переобучения.
- В метрических методах расстояния обычно становятся неинформативными при большом количестве признаков.



Неинформативность расстояний

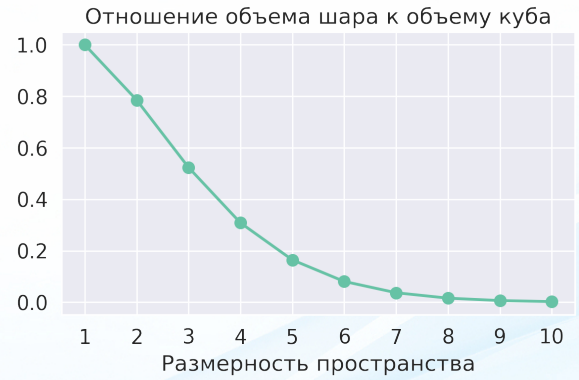
Объем шара радиуса 1 в $\mathbb{R}^d$

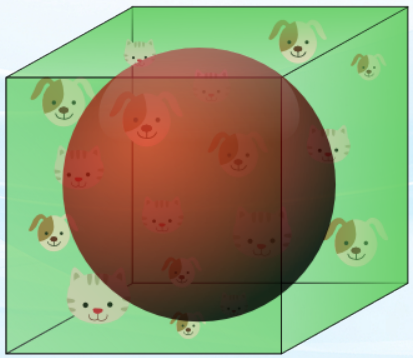
$$V_d = \frac{\pi^{d/2}}{\Gamma(d/2+1)}$$

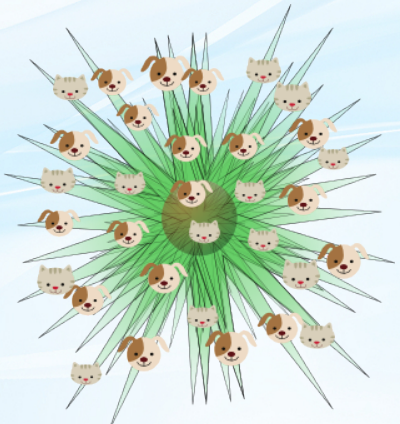


В пространстве большой размерности

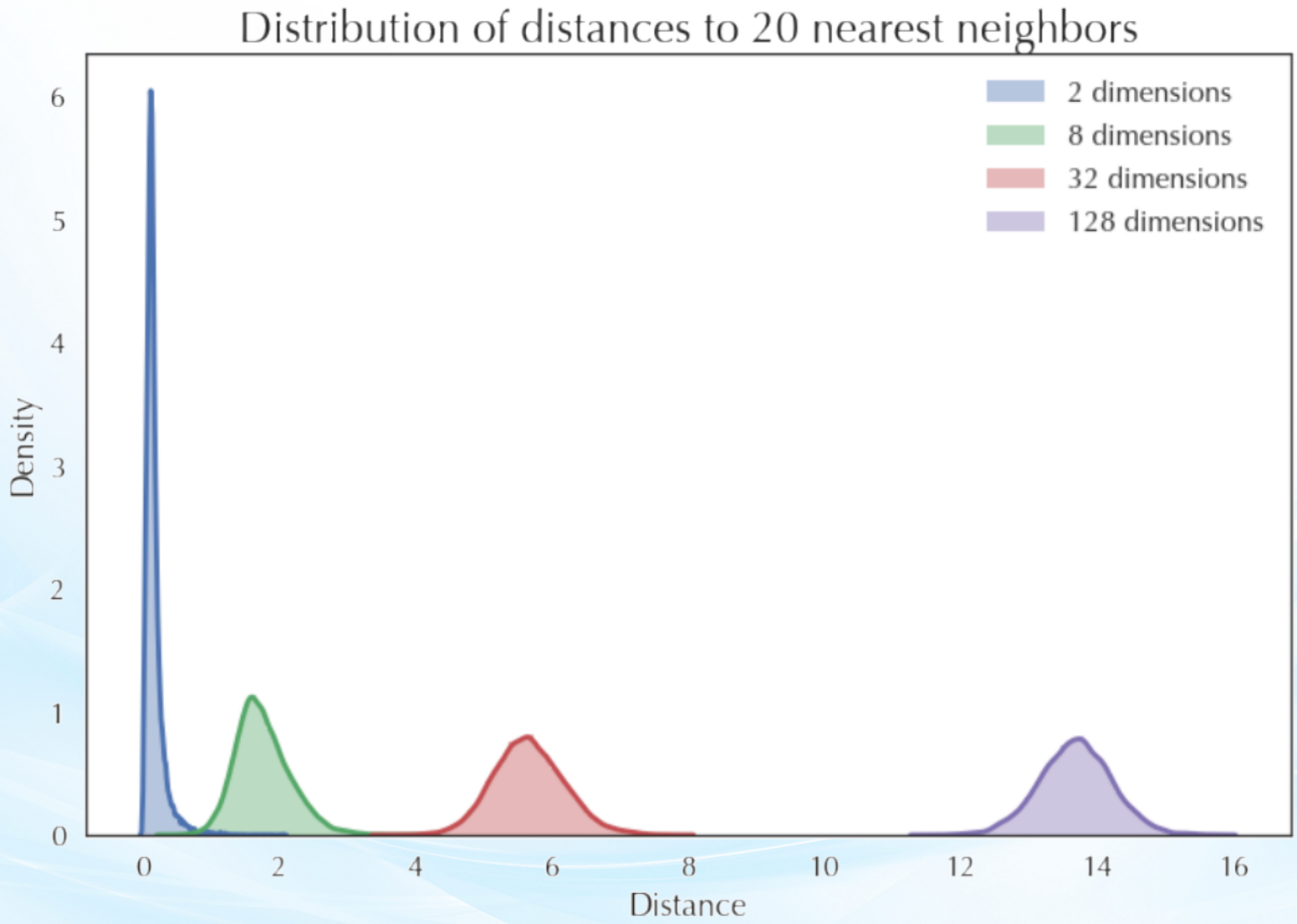
- объем шара сходится к нулю.
- почти весь объем шара сосредоточен вблизи его границы.



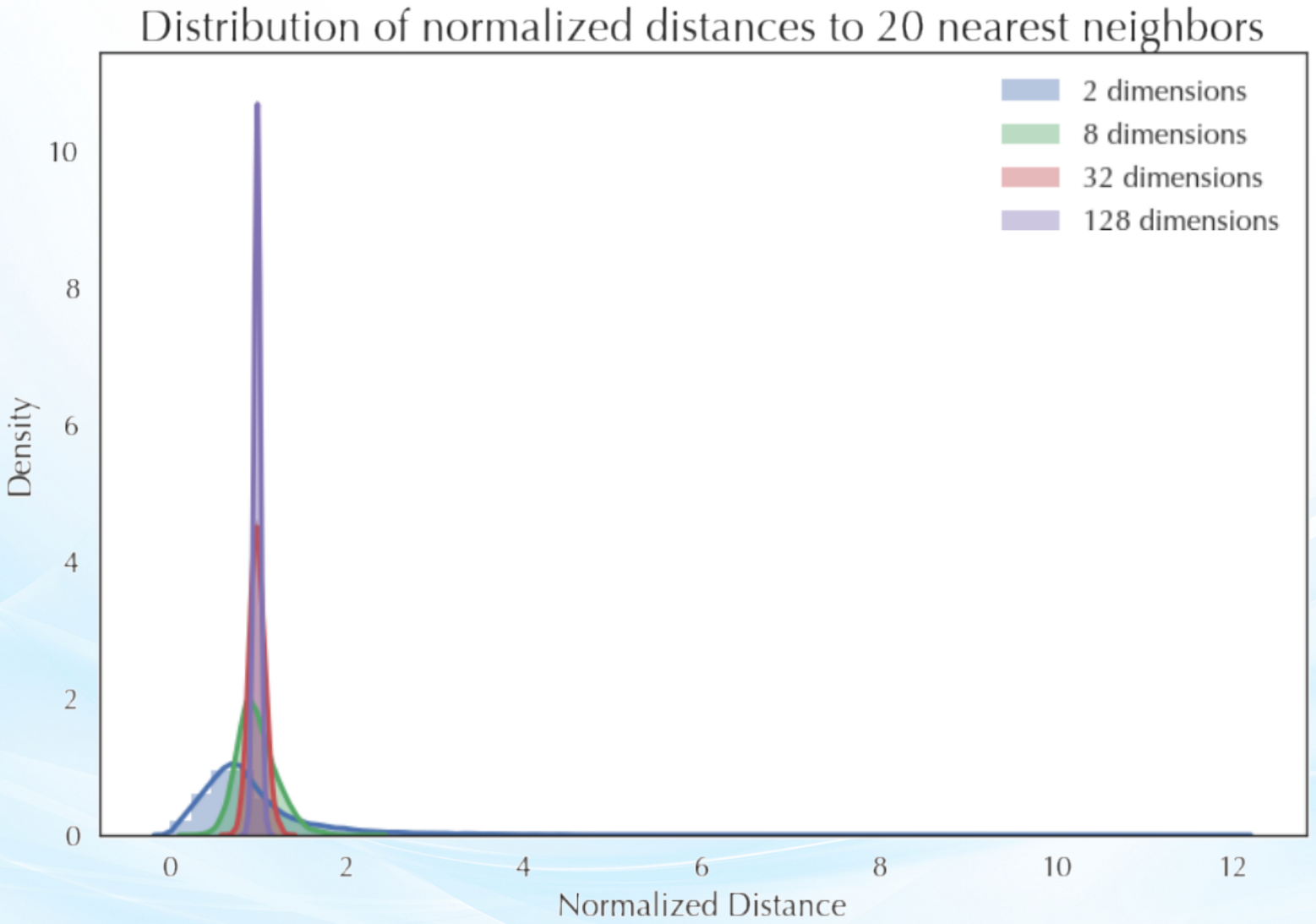




Распределение расстояний



Распределение нормированных расстояний



Метод PCA (Principal component analysis)

Дана выборка $X_1, ..., X_n \in \mathbb{R}^D$.

Задача: Найти подпространство размерности $d < D$

$L = \left\{ x \in \mathbb{R}^D \;\middle|\; x = x_0 + \sum_{j=1}^{d} y_j e_j, \quad y_j \in \mathbb{R} \right\}$,

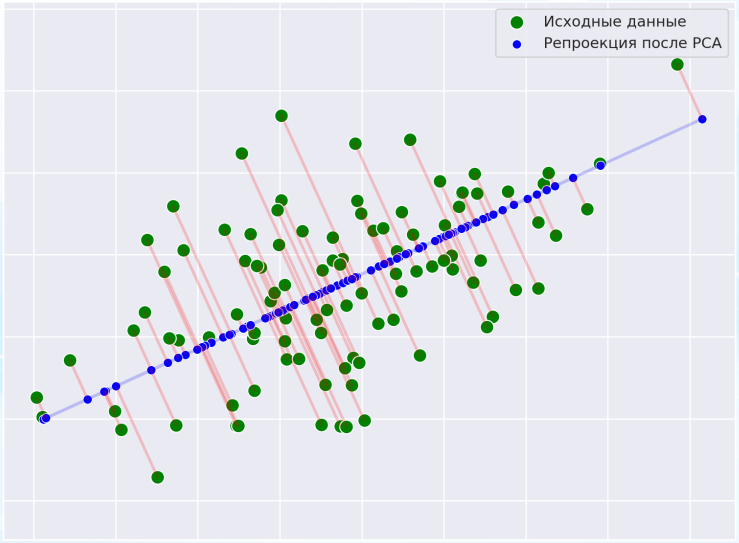
которое наилучшим образом приближает выборку.

Сразу центрируем данные, взяв $x_0 = \overline{X} := \frac{1}{n} \sum_{i=1}^{n} X_i$.

Заметим, что в этом подпространстве справедливо представление $x = x_0 + Sy$, где $S = (e_1, ..., e_d)$ — ортонормированный базис

Для матрицы данных получаем

$$X = \begin{pmatrix} x_1^T \\ ... \\ x_n^T \end{pmatrix} = x_0 + YS^T$$



Тем самым, для поиска наилучшего приближения нужно решить задачу

$$\left\| (X - x_0) - YS^T \right\|^2 \longrightarrow \min_{Y,S}$$

Метод PCA

Задача: $\left\|(X - x_0) - YS^T\right\|^2 \longrightarrow \min\limits_{Y,S}$

Решение задачи (см. линейная алгебра)

- S — матрица из d нормированных собственных векторов матрицы X^TX, соотв. наибольшим собств. значениям.
- $Y = (X - x_0)S$
- причем $S^TS = I_d$ и $Y^TY = diag(\lambda_1, ..., \lambda_d)$
- $\left\|(X - x_0) - YS^T\right\|^2 = \sum\limits_{j=d+1}^{D} \lambda_j$, т.е. наименьшие собств. знач.

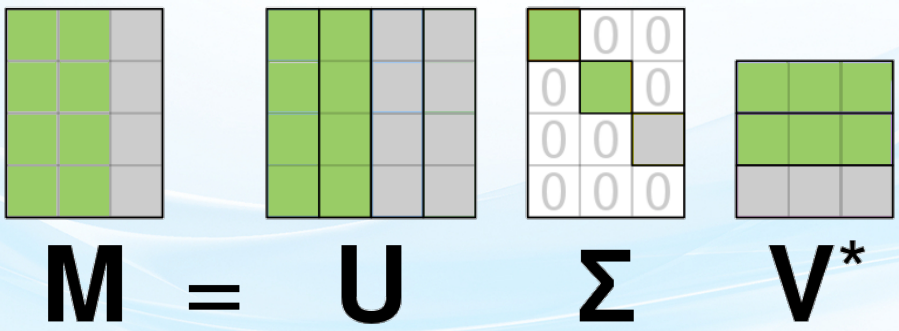
Итоговая схема

1. *Переход в пространство малой размерности*
   Имеется исходный объект $x \in \mathbb{R}^D$
   Его проекция $y = S^T(x - x_0) \in \mathbb{R}^d$
2. *Обратный переход в пространство большой размерности*
   Имеется проекция $y \in \mathbb{R}^d$
   Восстановленный $\widehat{x} = Sy + x_0 \in \mathbb{R}^D$, возможна потеря данных

Метод PCA

Теорема SVD. Произвольная $n \times m$ матрица M представима в виде сингулярного разложения $M = U\Sigma V^T$, где

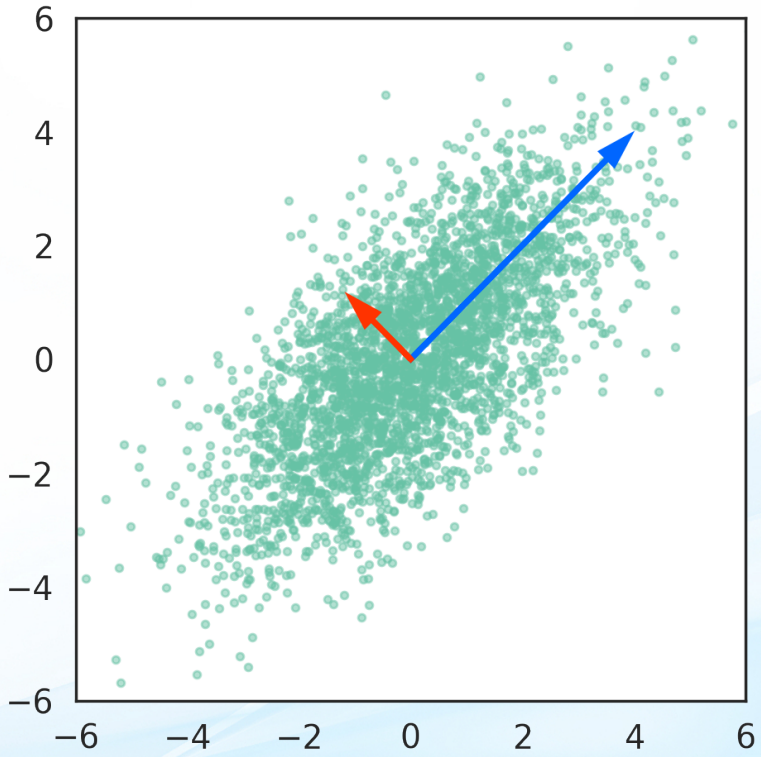
1. $U = (u_1, ..., u_m)$ ортогональна, $U^T U = I_m$,
   u_j — собственные векторы MM^T;
2. $V = (u_1, ..., u_m)$ ортогональна, $V^T V = I_m$,
   v_j — собственные векторы $M^T M$;
3. $\Sigma = \text{diag}(\lambda_1, ..., \lambda_m), \lambda_i \geqslant 0$ — собственные числа $M^T M$ и MM^T.



Для получения PCA фактически остается "обрезать" матрицы.

Метод PCA

- Вычислять можно с помощью SVD-разложения.
- Является линейным методом, т.к. получает линейное подпр-во.
- Устойчив к проклятию размерности.
- Получается проекция данных на те направления, по которым наблюдается наибольшая дисперсия.



ВСЁ!