



Академия
Аналитиков
Авито

Кластеризация и понижение размерности

Академия Аналитиков Авито

Сакаев Руслан, 2023

ПОВТОРЕНИЕ

- Как работают сверточные нейронные сети
- Как устроены convolutional и pooling слои
- Изучили некоторые сверточные архитектуры и решаемые задачи

TECT



<https://anketolog.ru/s/779608/BUxGREvM>

ПЛАН ЛЕКЦИИ

Тест

- Обучение без учителя: кластеризация
- Алгоритмы кластеризации:
 - k-Means
 - EM algorithm
 - DBScan
- Оценка кластеризации

Перерыв

- Обучение без учителя: задача понижения размерности
- Алгоритмы понижения размерности:
 - PCA
 - t-SNE
- Выводы

СОДЕРЖАНИЕ

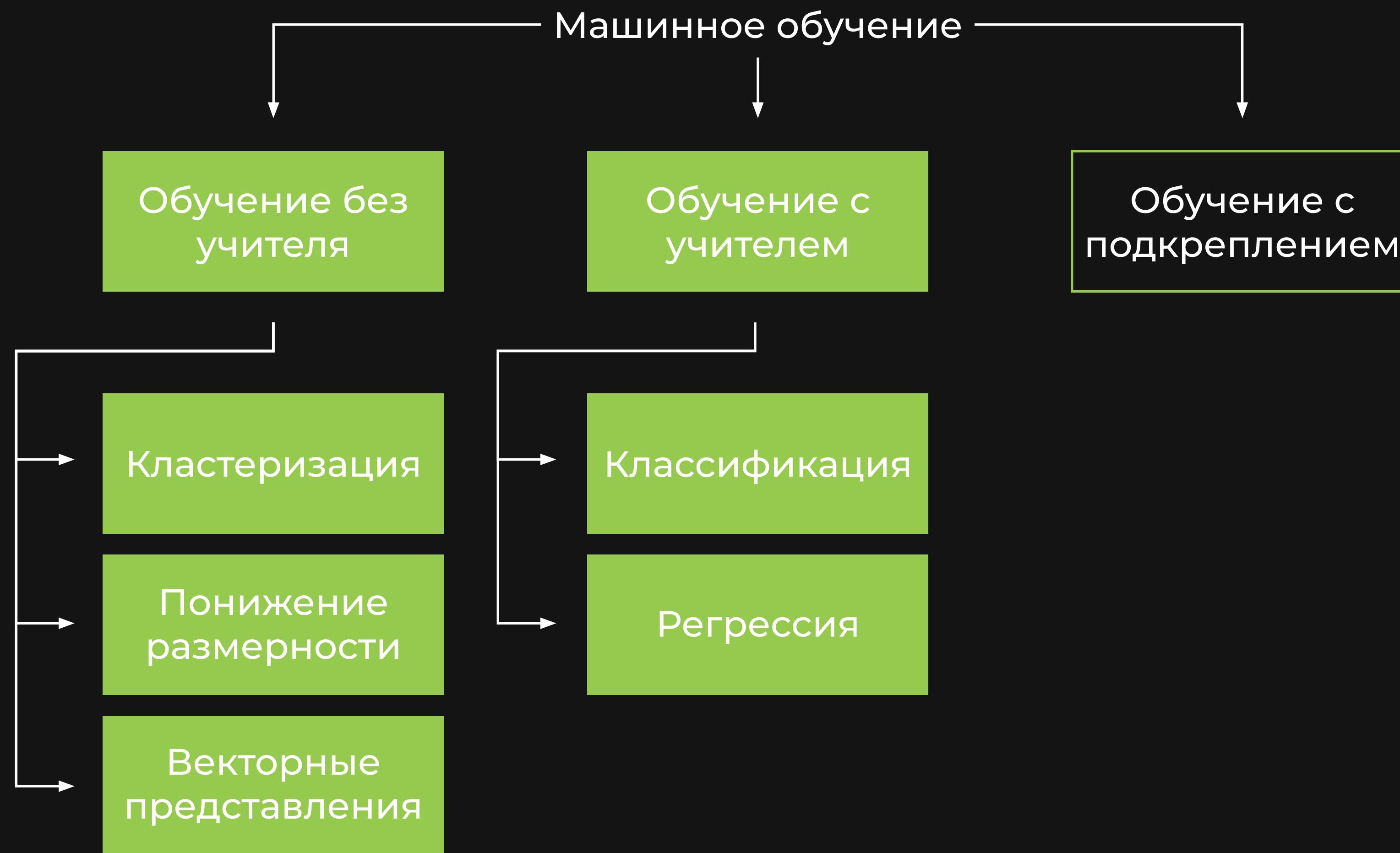
- Обучение без учителя:
кластеризация
- Алгоритмы кластеризации:
 - k-Means
 - EM algorithm
 - DBScan
- Оценка кластеризации

СОДЕРЖАНИЕ

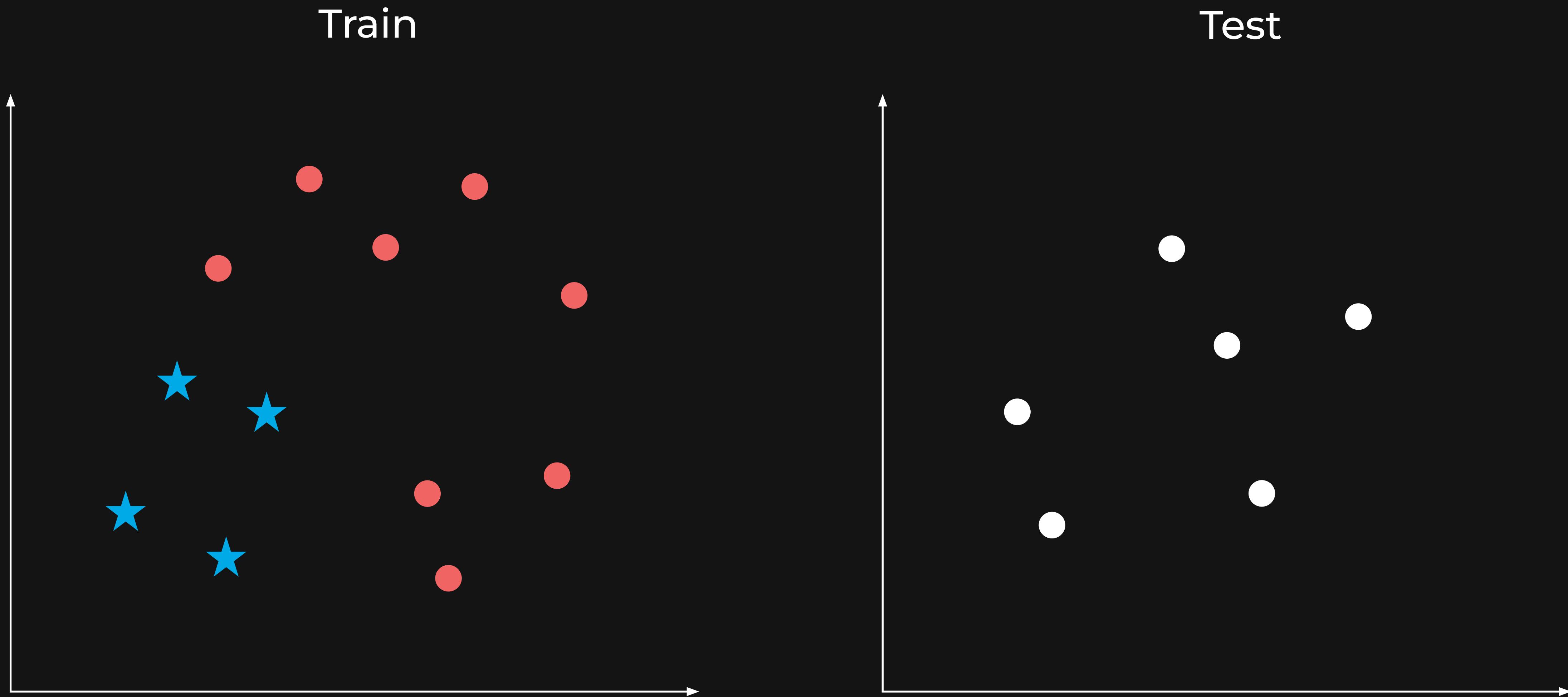
- Обучение без учителя:
кластеризация
- Алгоритмы кластеризации:
 - k-Means
 - EM algorithm
 - DBScan
- Оценка кластеризации



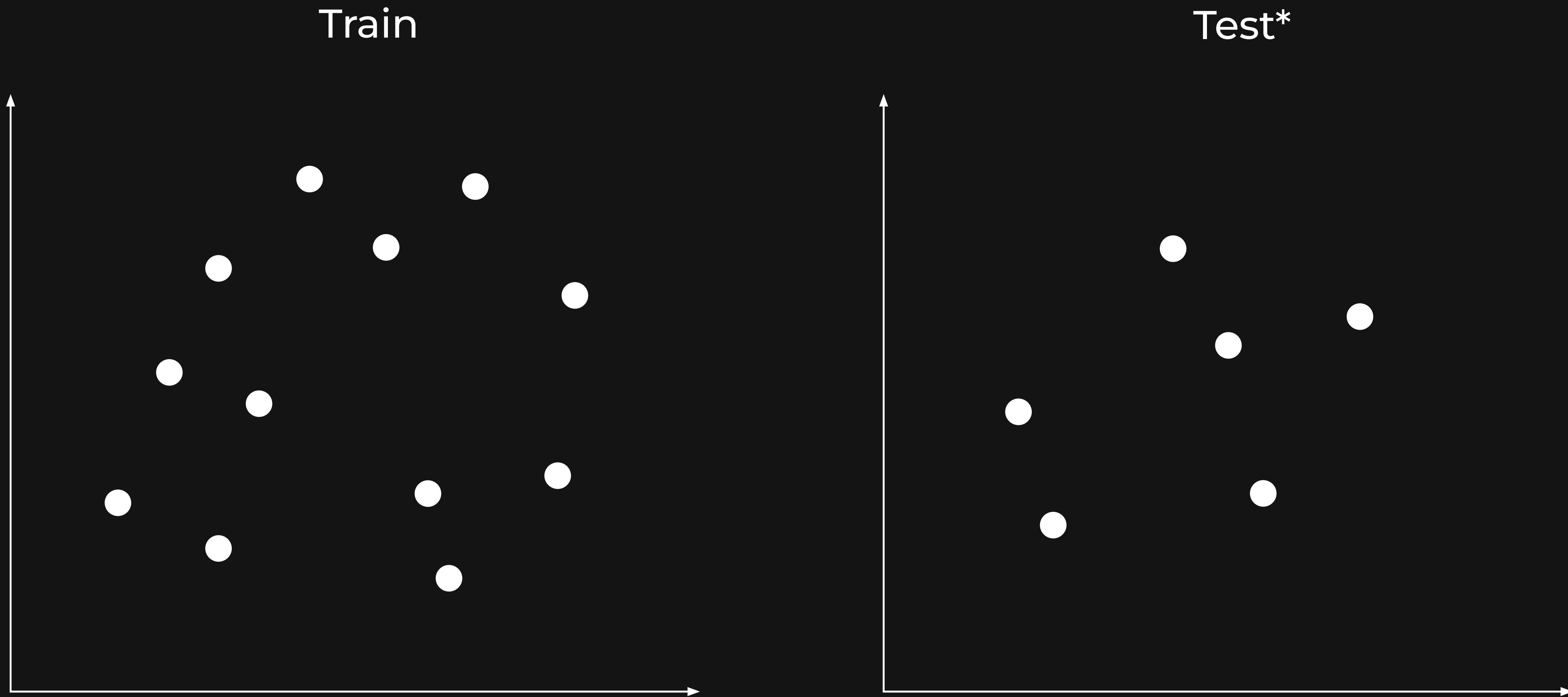
МАШИННОЕ ОБУЧЕНИЕ



ОБУЧЕНИЕ С УЧИТЕЛЕМ: КЛАССИФИКАЦИЯ



ОБУЧЕНИЕ С УЧИТЕЛЕМ: КЛАССТЕРИЗАЦИЯ



ПРИМЕРЫ ЗАДАЧ КЛАСТЕРИЗАЦИИ: ПОЛЬЗОВАТЕЛИ



Пользователь 1

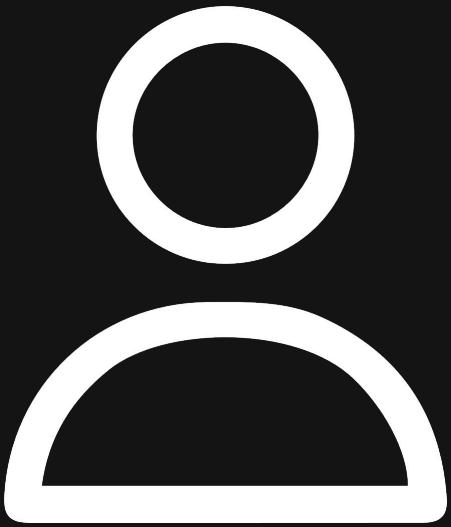
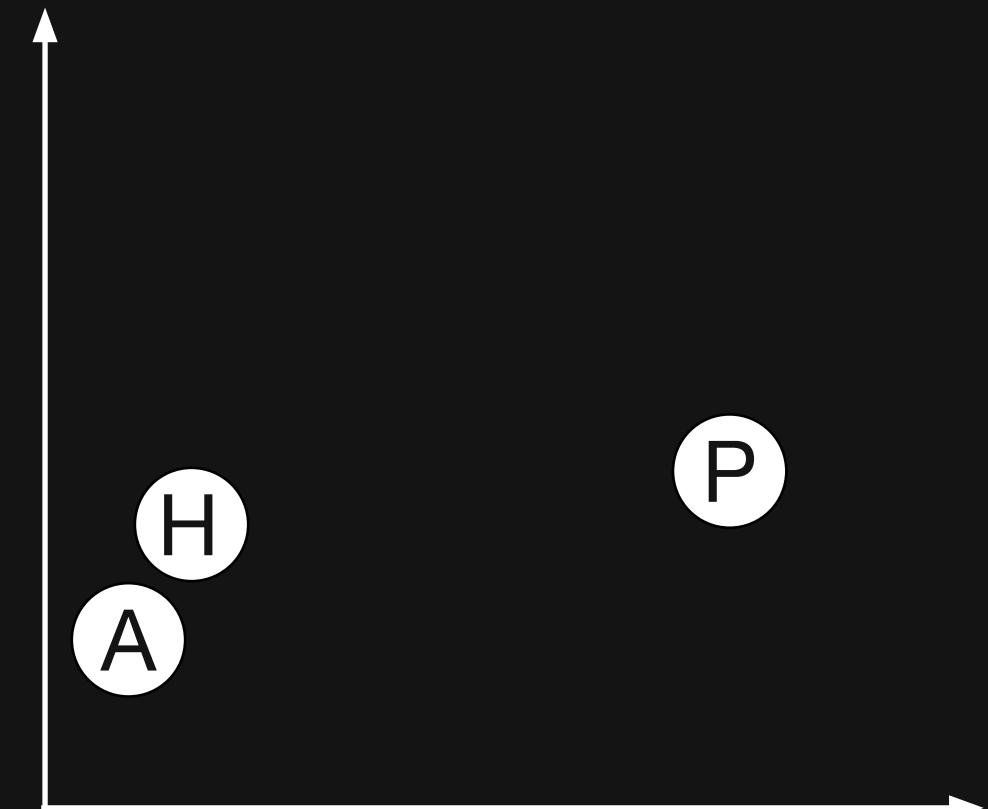
- ▶ Устройство
- ▶ Сеть
- ▶ Поведение

H

Аккаунт Николай

A

Аккаунт Анастасия



Пользователь 2

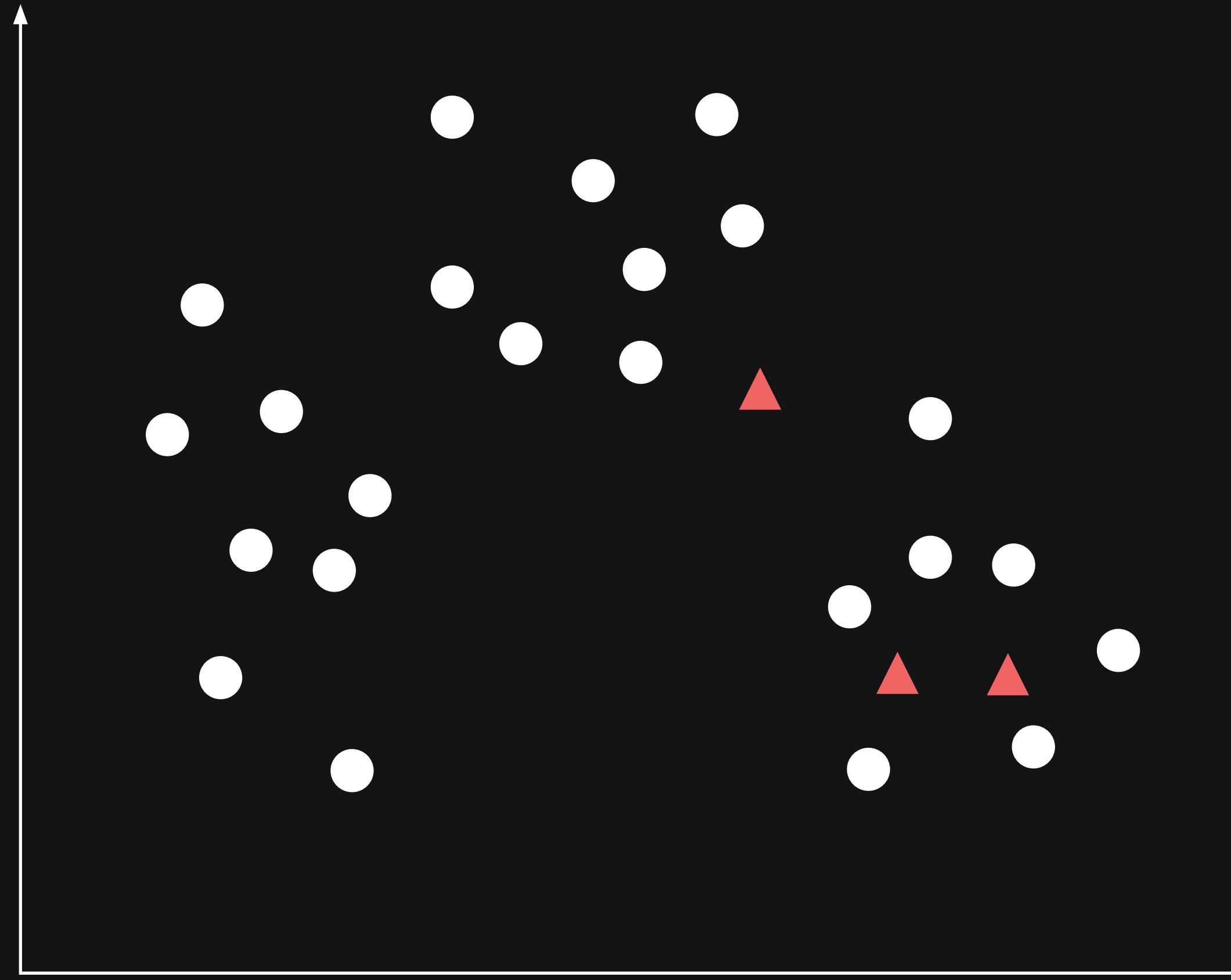
- ▶ Устройство
- ▶ Сеть
- ▶ Поведение

P

Аккаунт Роман

ПРИМЕРЫ ЗАДАЧ КЛАСТЕРИЗАЦИИ: ТЕКСТ

Train



интересен Ваш автомобиль. Могу
купить его за девяноста % от его
рыночной цены. Наберите меня 8
(9,2,6) 329-33-67 19:57✓

По автомобилю кидываешь цену?
Могу срочно забрать сегодня до 90
процентов от стоимости. Набери
меня обсудим +7 (902)830-8-881
19:58✓

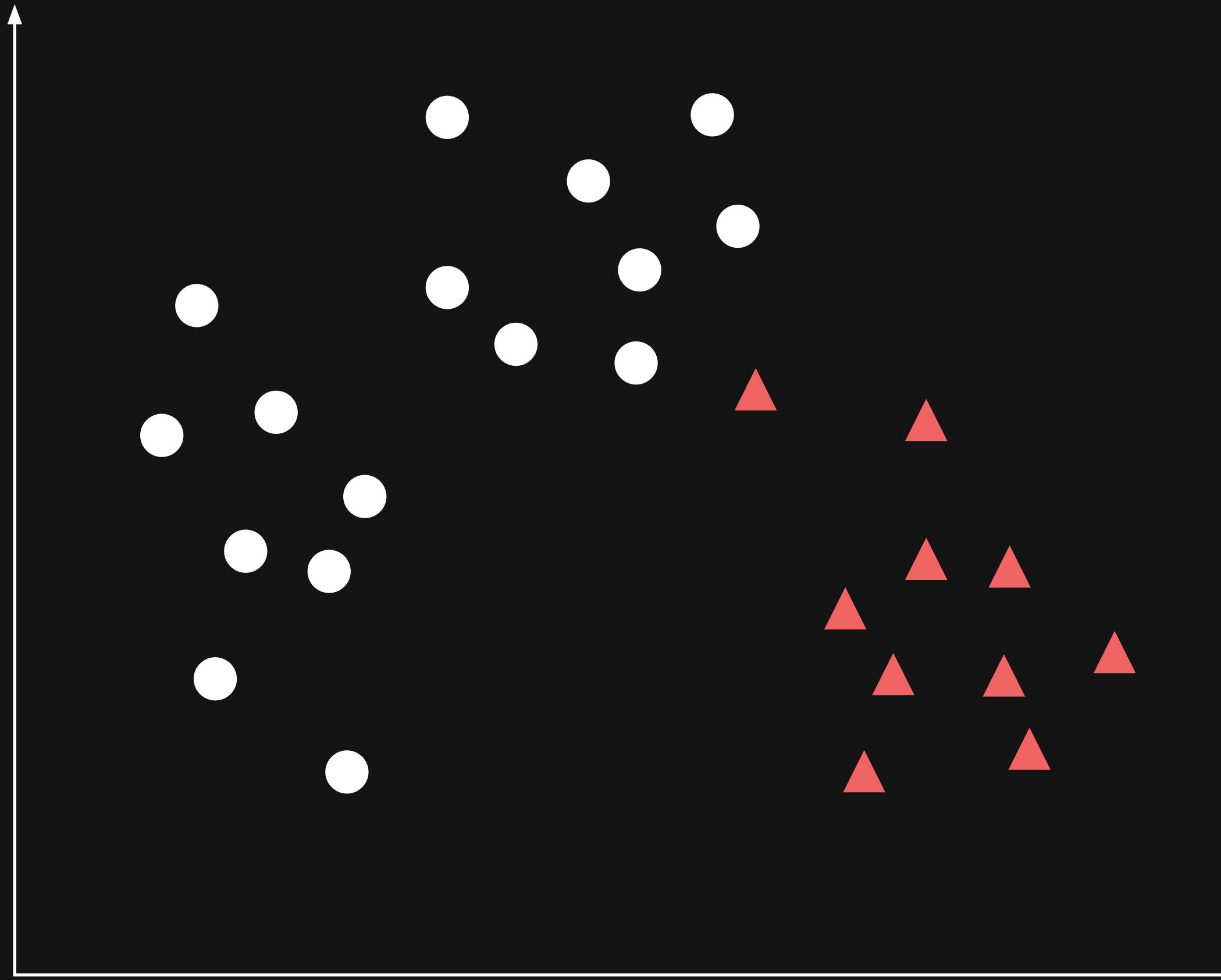
Хай. Мне требуется в Калугу,
ответьте пожалуйста.Wts/AP 19:58✓

980512 19:58✓

5114 19:58✓

ПРИМЕРЫ ЗАДАЧ КЛАСТЕРИЗАЦИИ: ТЕКСТ

Test



интересен Ваш автомобиль. Могу
купить его за девяноста % от его
рыночной цены. Наберите меня 8
(9,2,6) 329-33-67 19:57✓

По автомобилю кидываешь цену?
Могу срочно забрать сегодня до 90
процентов от стоимости. Набери
меня обсудим +7 (902)830-8-881
19:58✓

Хай. Мне требуется в Калугу,
ответьте пожалуйста.Wts/AP 19:58✓

980512 19:58✓

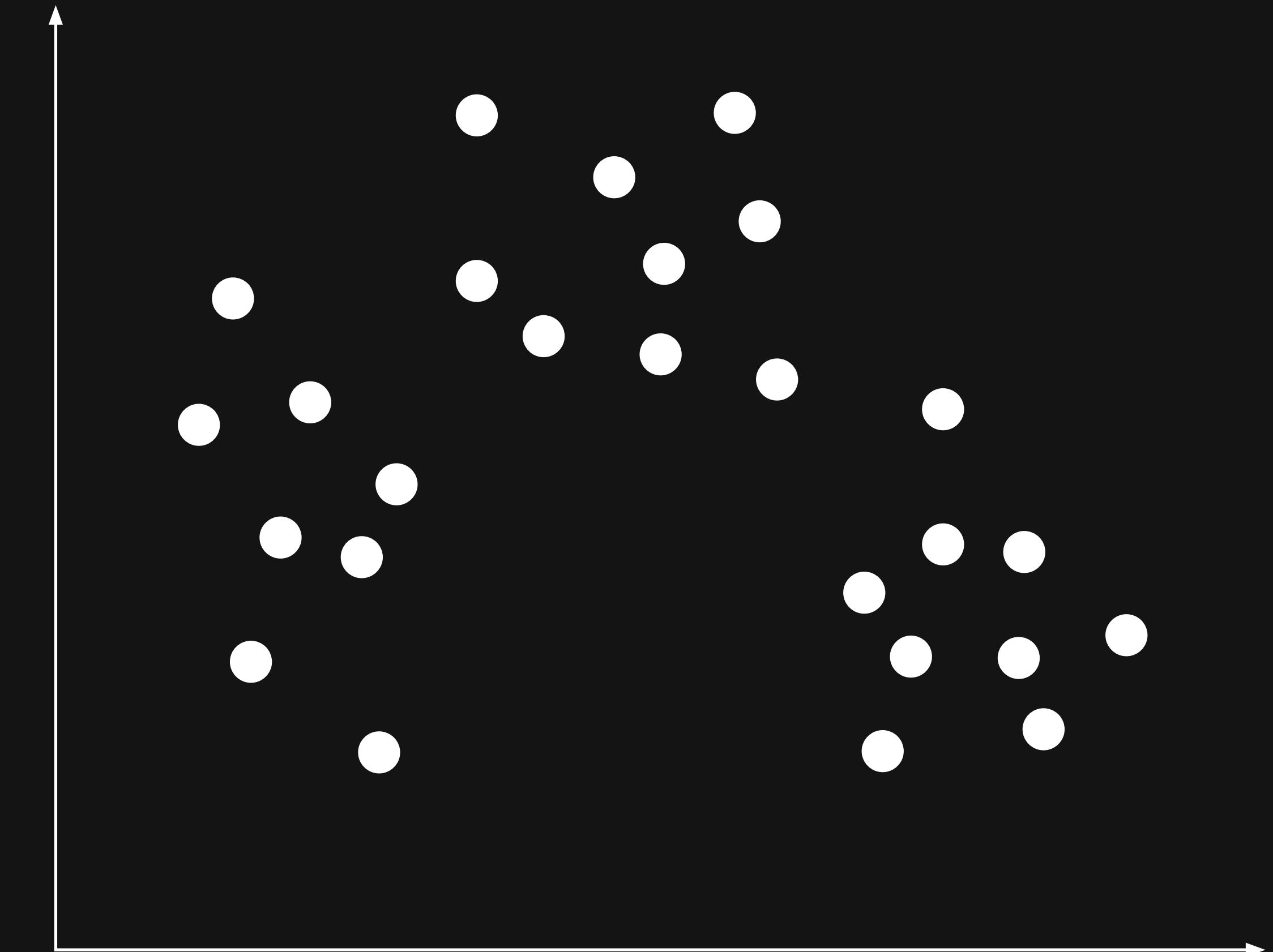
5114 19:58✓

ПРИМЕРЫ ЗАДАЧ КЛАСТЕРИЗАЦИИ: УГЛУБЛЕНИЕ КАТЕГОРИЗАЦИИ



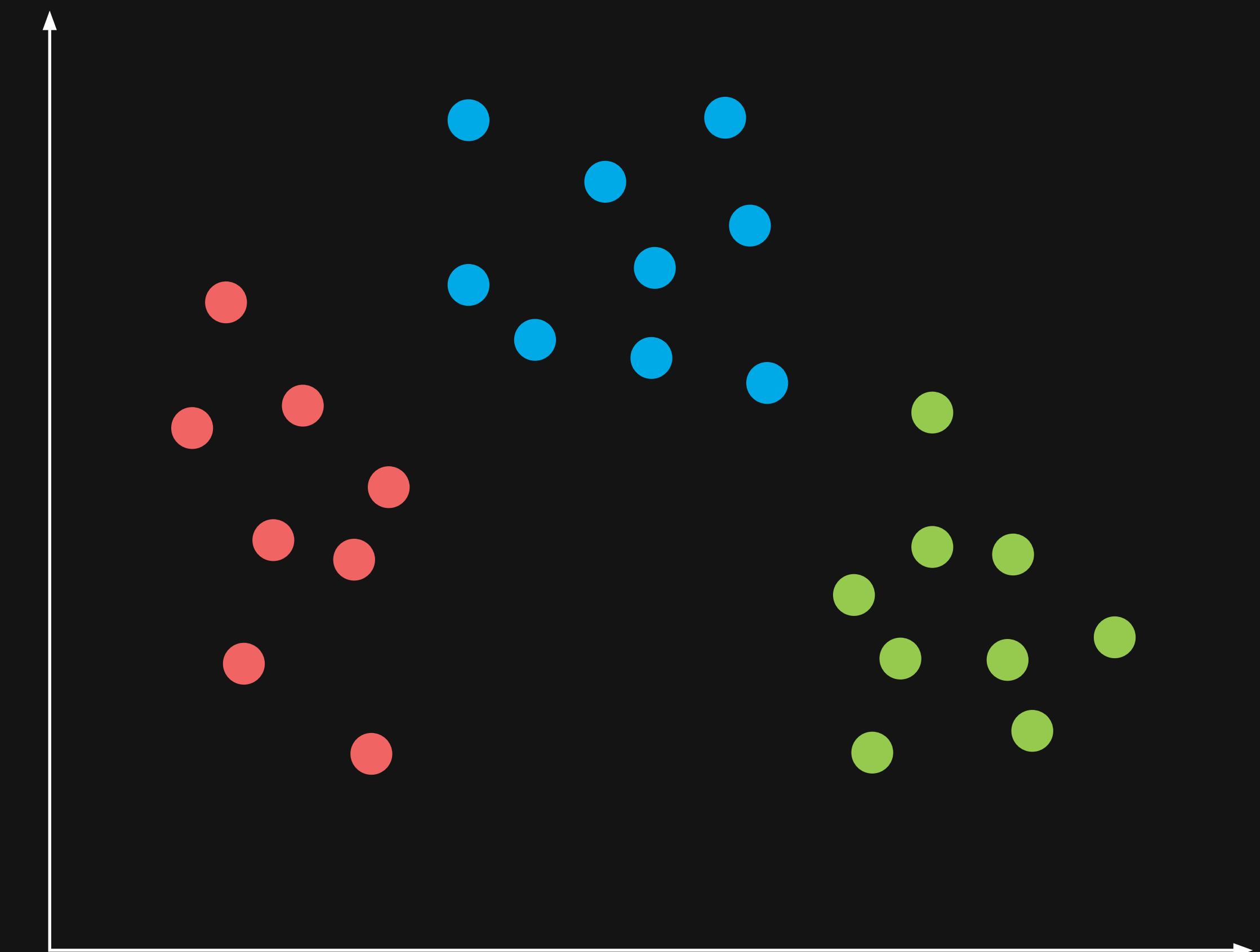
ПОСТАНОВКА ЗАДАЧИ КЛАСТЕРИЗАЦИИ

- **Дано:**
Имеется обучающая выборка X и функция
расстояния между объектами ρ .
- **Задача:**
 - Разбить выборку на непересекающиеся
подмножества, называемые **кластерами**
 - Каждый кластер состоит из объектов,
близких по метрике ρ
 - Объекты разных кластеров существенно
отличаются



НЕОДНОЗНАЧНОСТЬ РЕШЕНИЯ

- не существует однозначно наилучшего критерия качества кластеризации
- число кластеров неизвестно заранее
- результат кластеризации существенно зависит от метрики расстояния



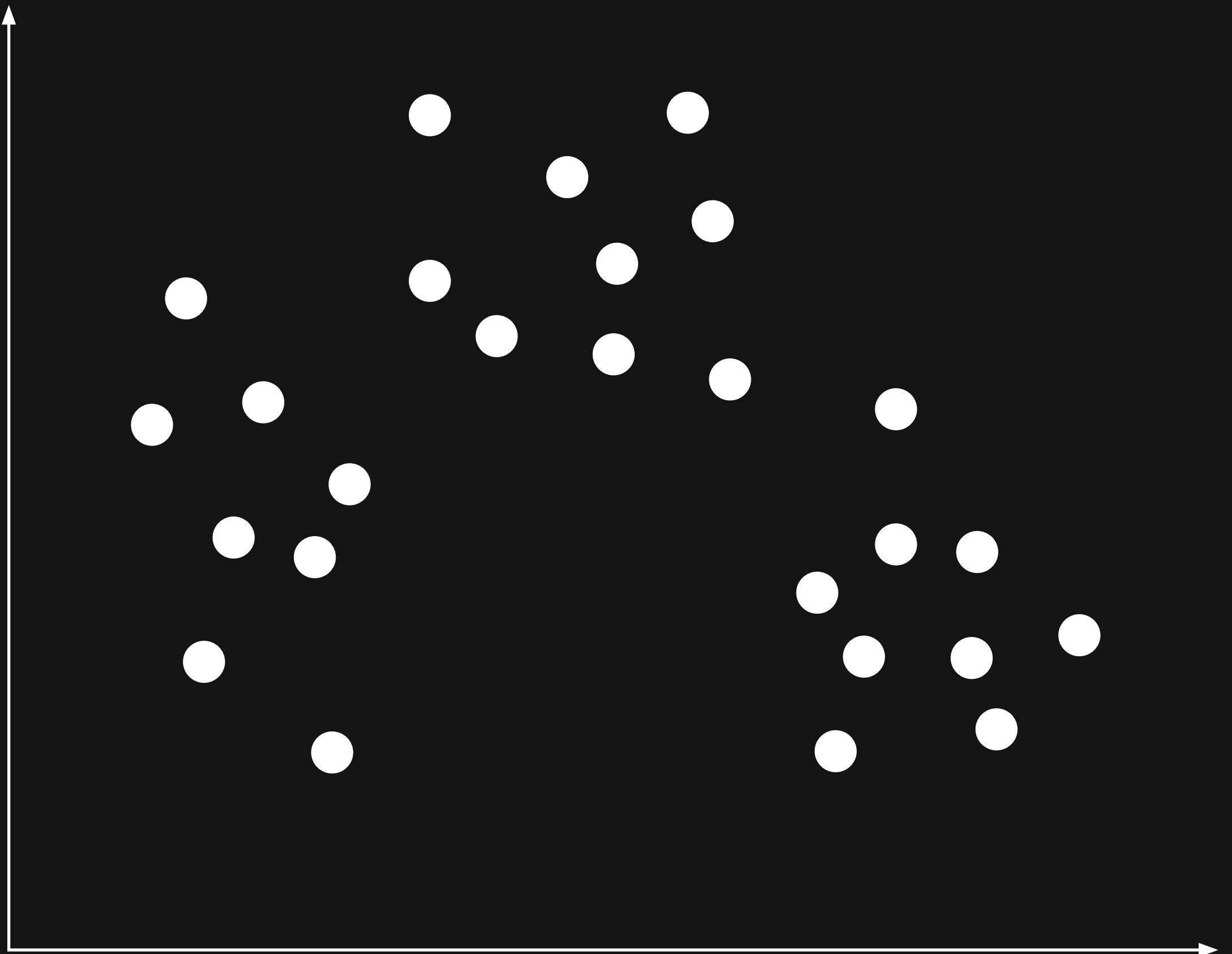
СОДЕРЖАНИЕ

- Обучение без учителя:
кластеризация
- Алгоритмы кластеризации:
 - k-Means
 - EM
 - DBScan
- Оценка кластеризации

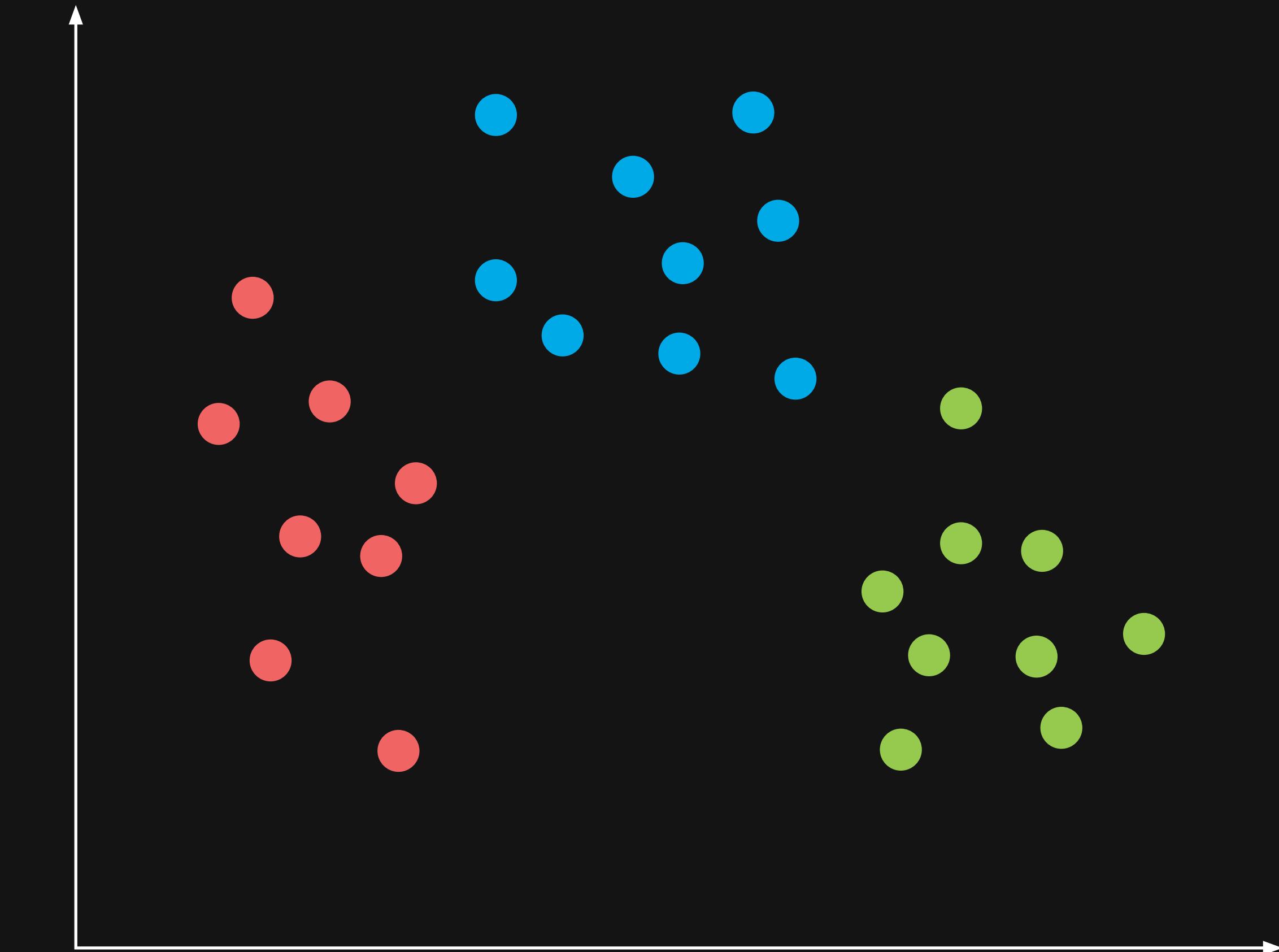
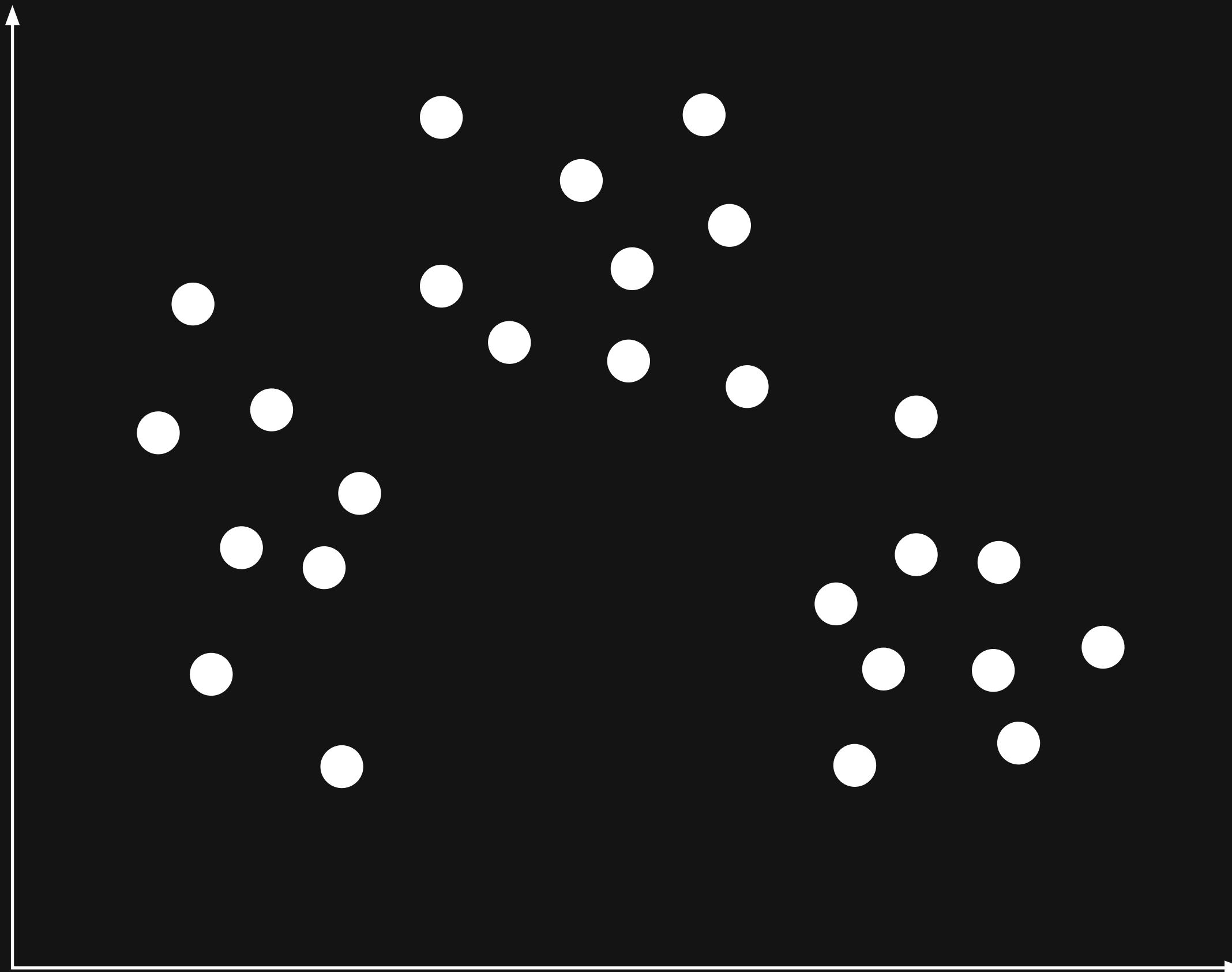
K-MEANS BE LIKE



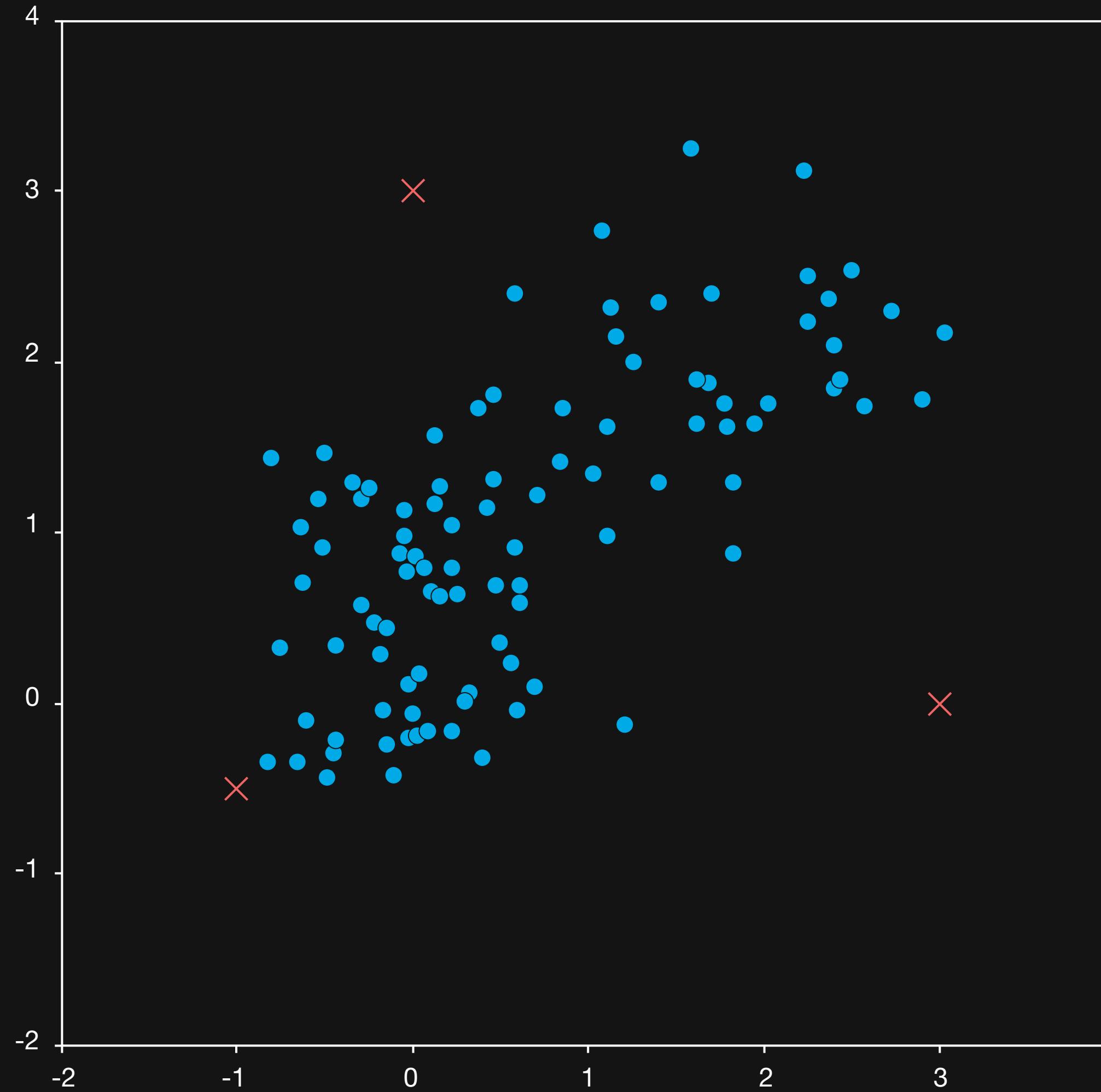
ЖЁСТКАЯ КЛАСТЕРИЗАЦИЯ



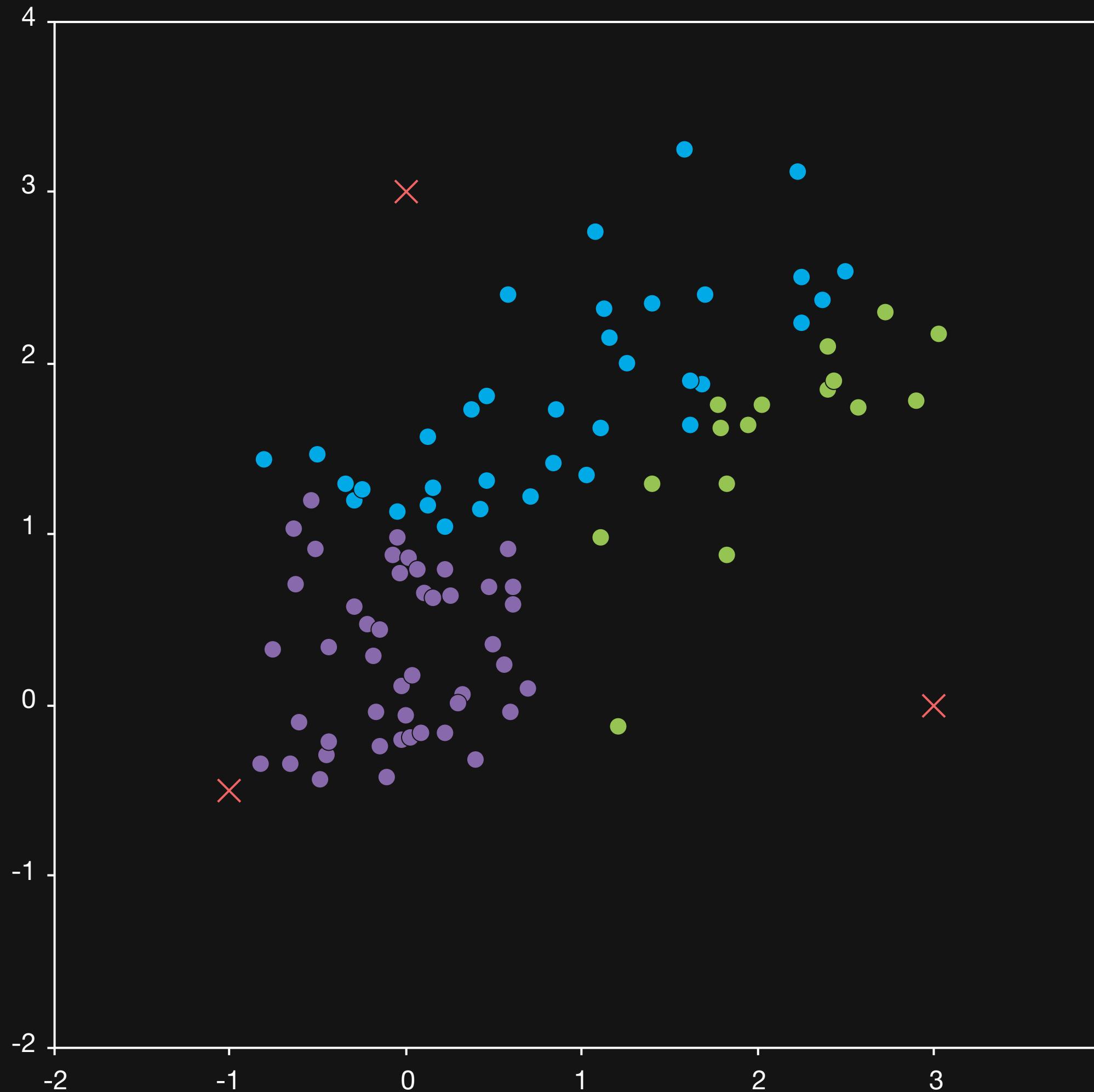
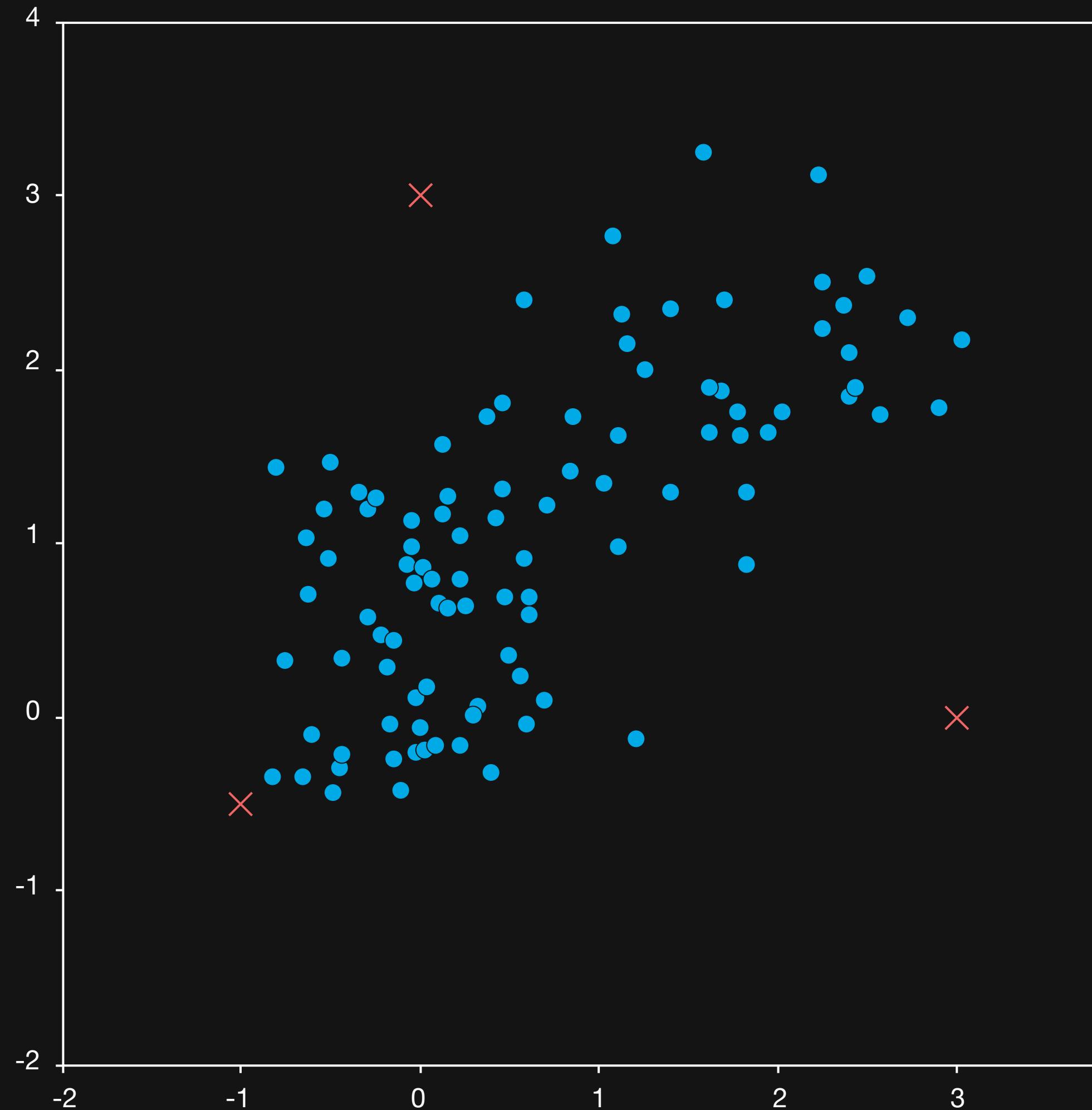
ЖЁСТКАЯ КЛАСТЕРИЗАЦИЯ



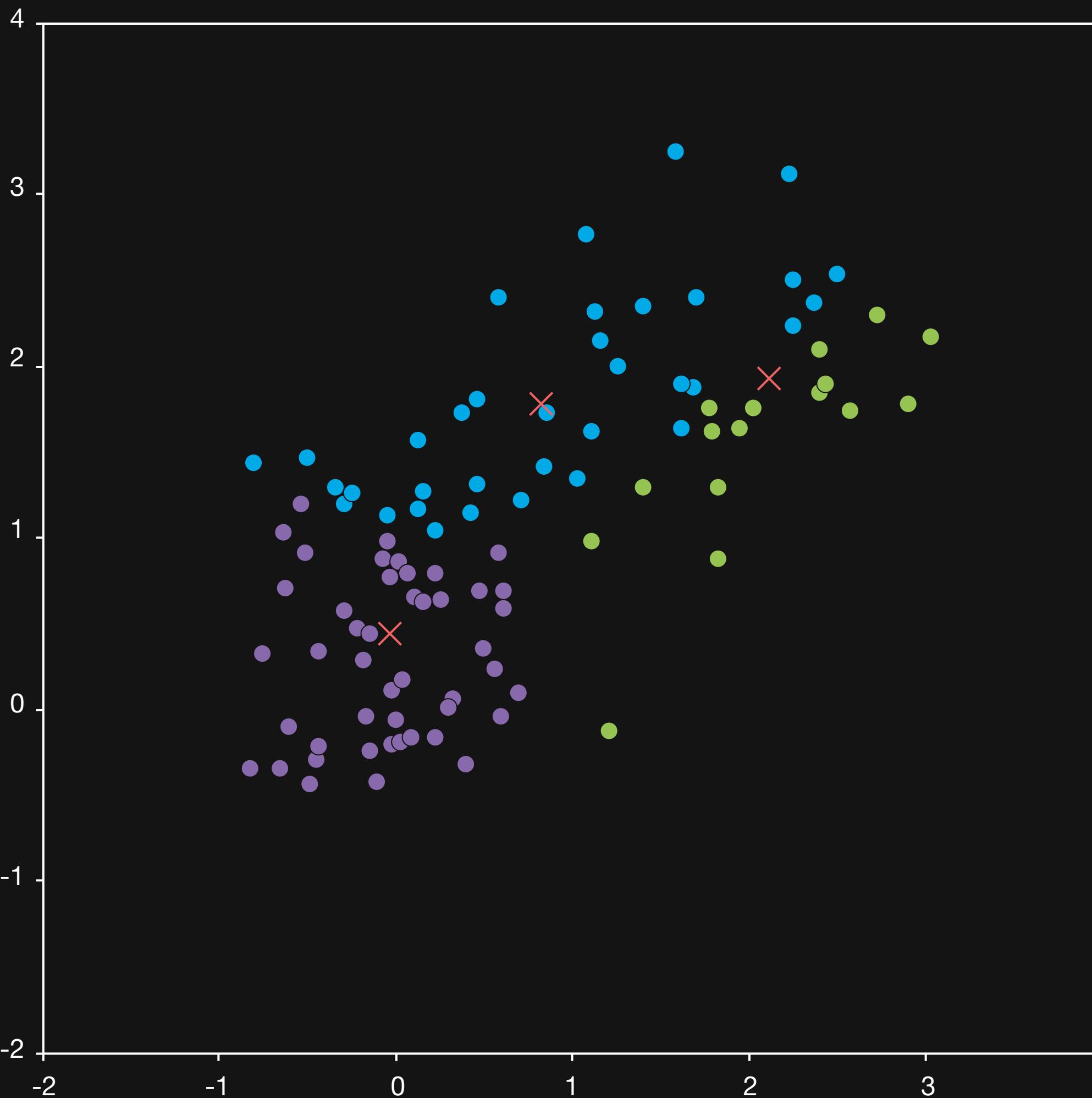
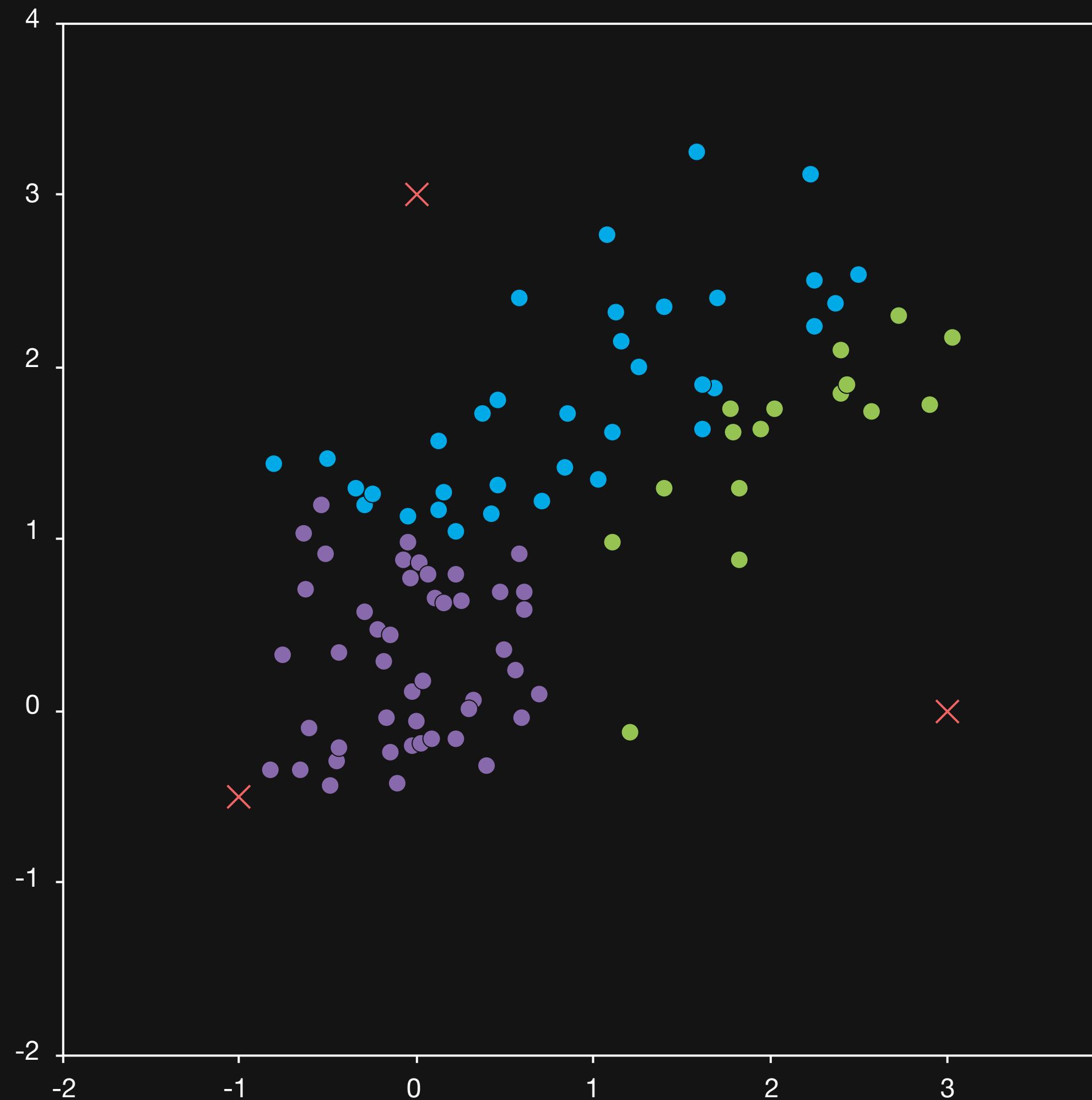
KMEANS: 3 КЛАСТЕРА



KMEANS: 3 КЛАСТЕРА



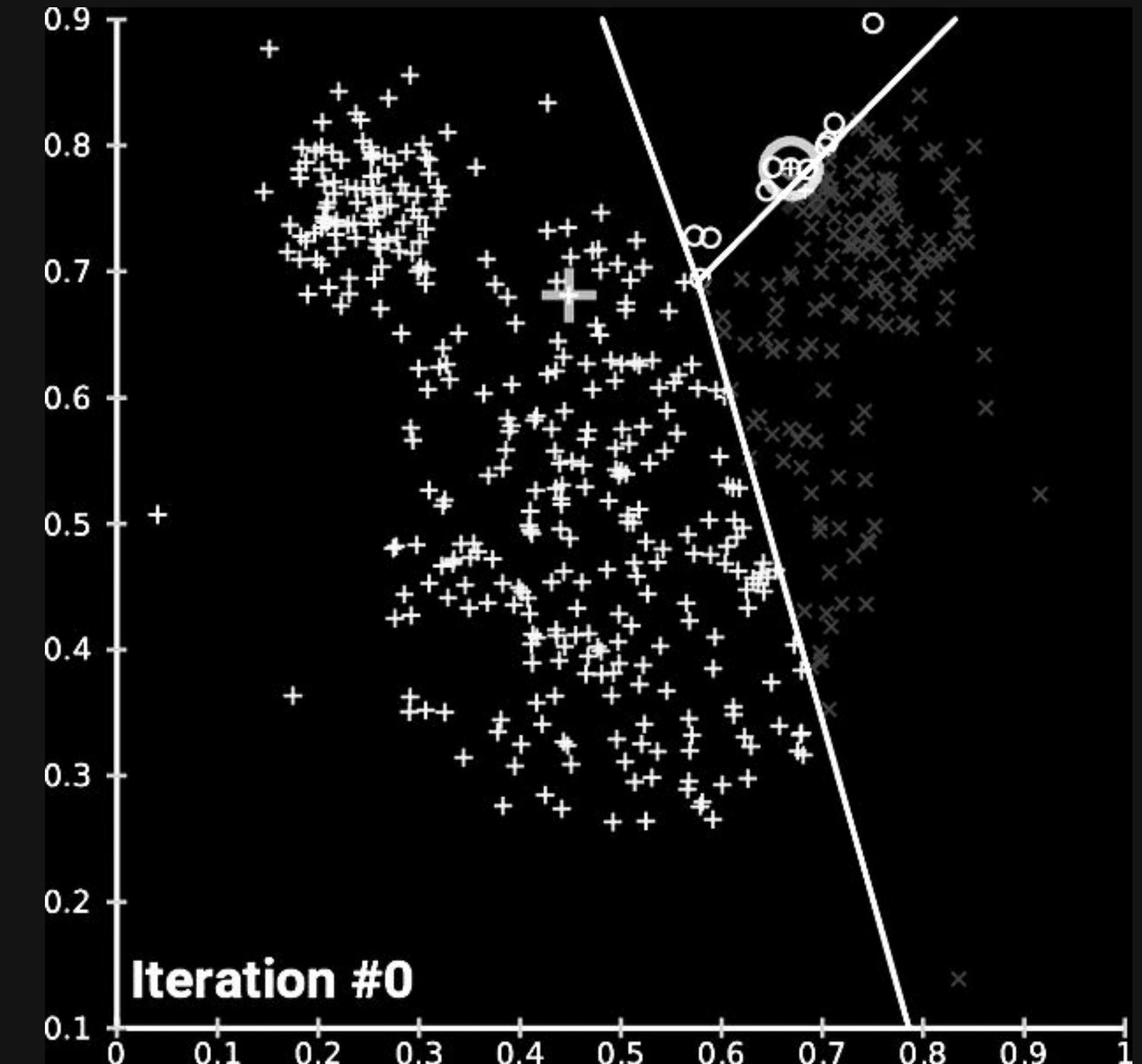
KMEANS: 3 КЛАСТЕРА



KMEANS

- Инициализировать центры кластеров
- Отнести точки к ближайшим кластерам
- Пересчитать положение центров
- Повторять до сходимости

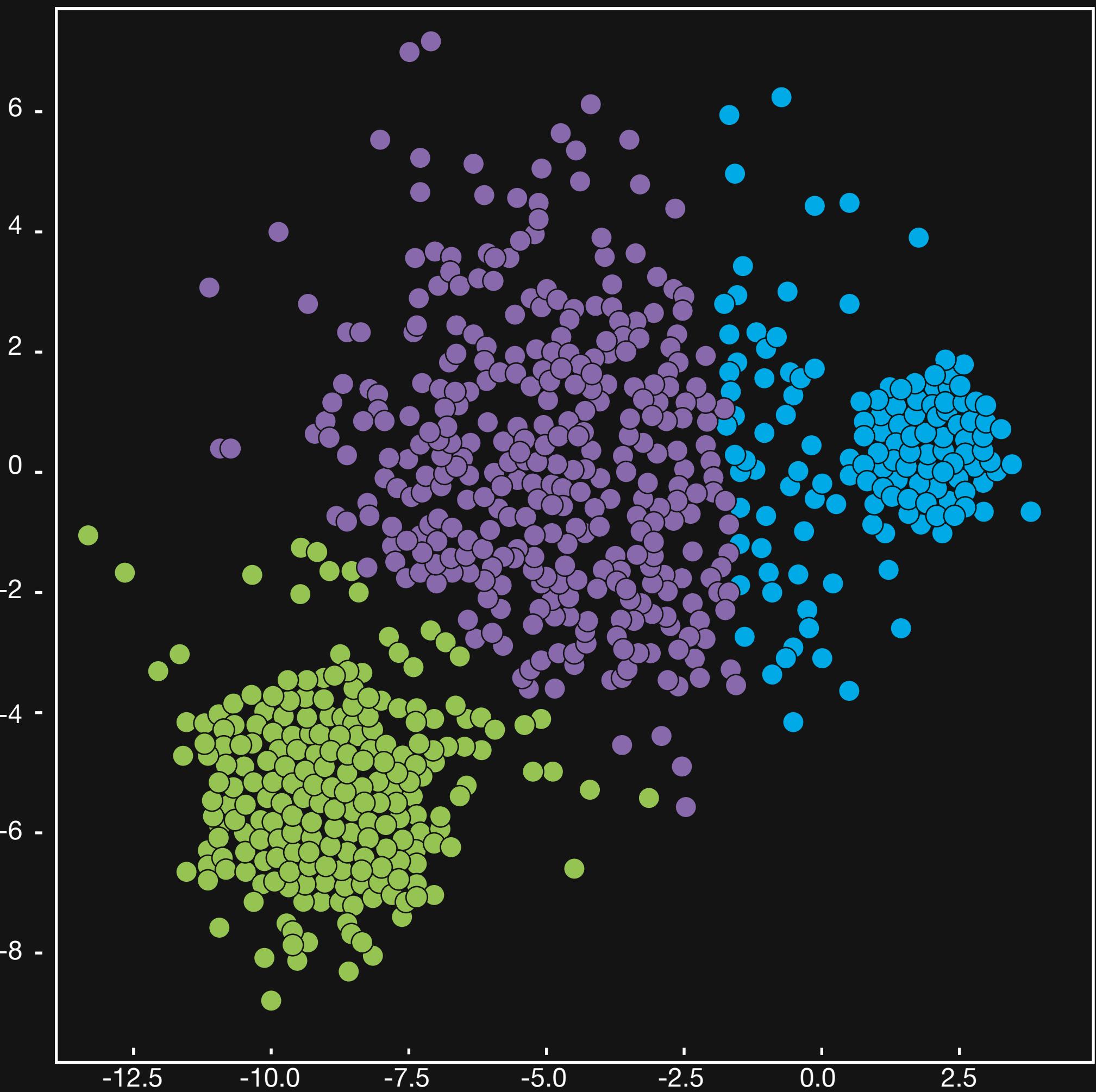
$$V = \sum_{i=1}^k \sum_{x \in S_i} (x - \mu_i)^2$$



ОГРАНИЧЕНИЯ KMEANS

- Кластеры могут быть только выпуклыми
- Кластеры должны быть с одинаковой дисперсией
- Примерно равное количество элементов в кластерах
- Не слишком большое количество кластеров

Kmeans для кластеров с разной дисперсией



KMEANS

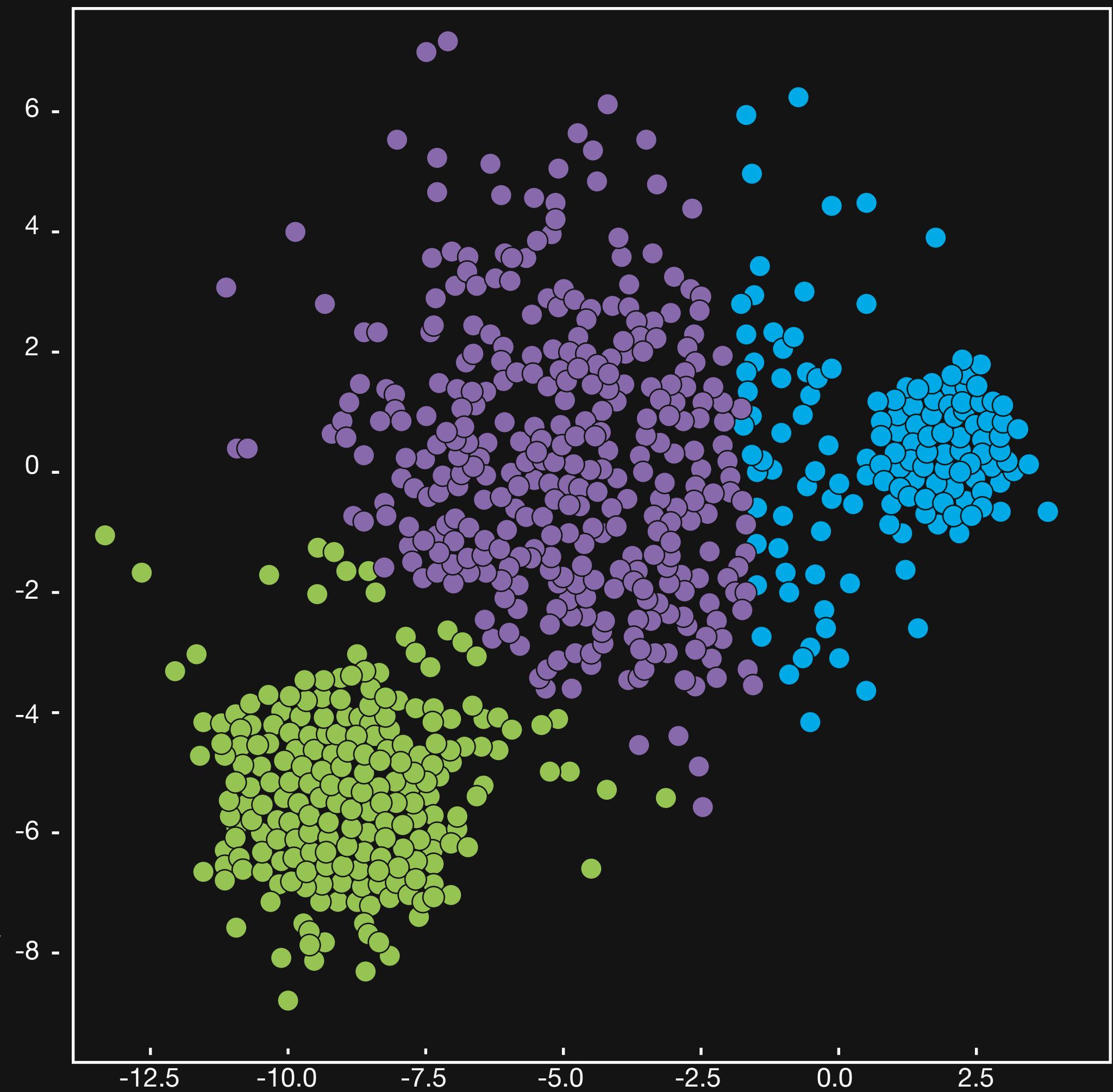
Плюсы:

- Простой и быстрый
- Неплохое качество
- Множество модификаций

Минусы:

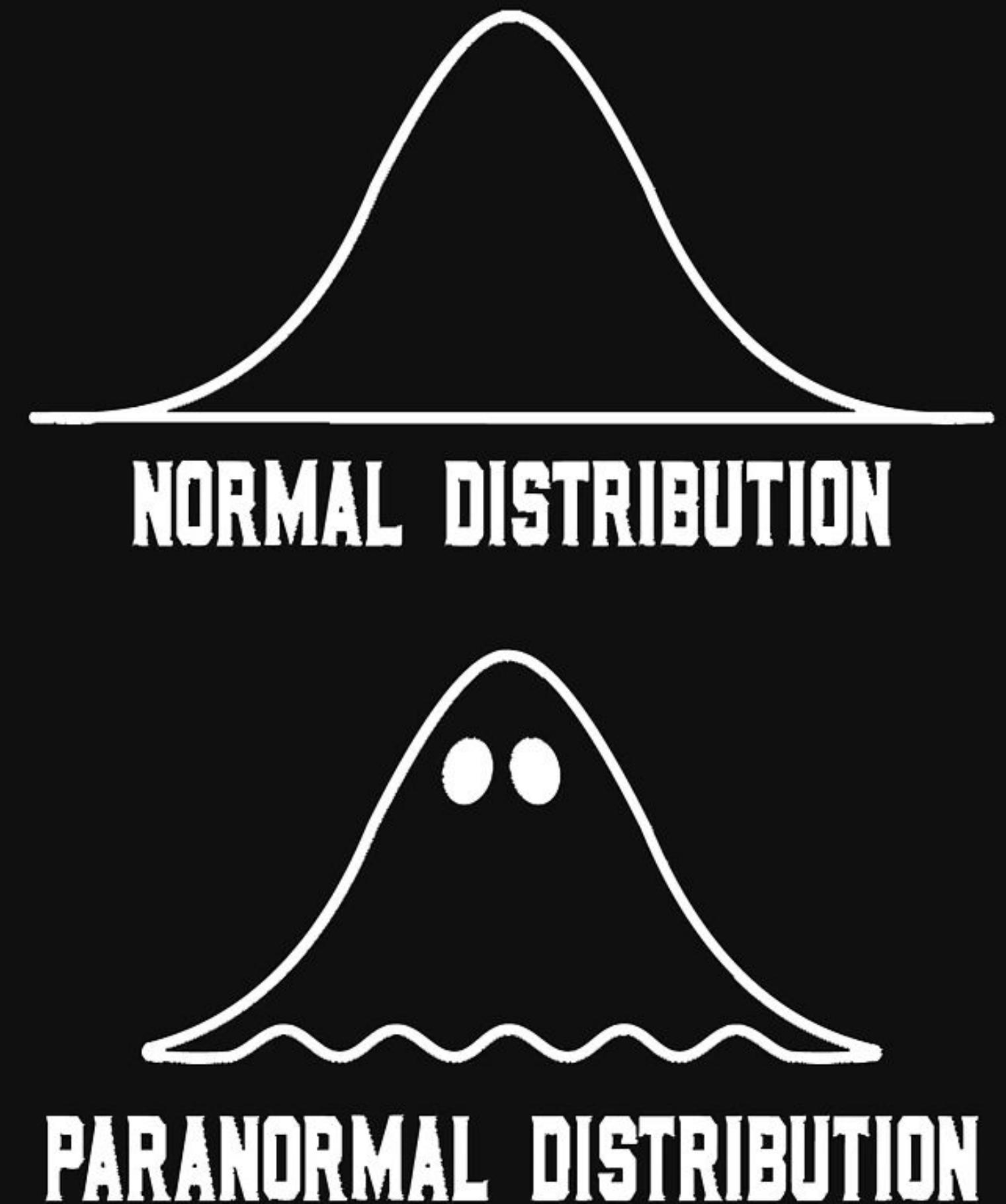
- Нужно выбирать количество кластеров
- Нужно нормировать данные
- Чувствительность к начальным условиям
- Чувствительность к выбросам и шумам
- Возможность сходимости к локальному оптимуму
- Жесткие ограничения

Kmeans для кластеров с разной дисперсией

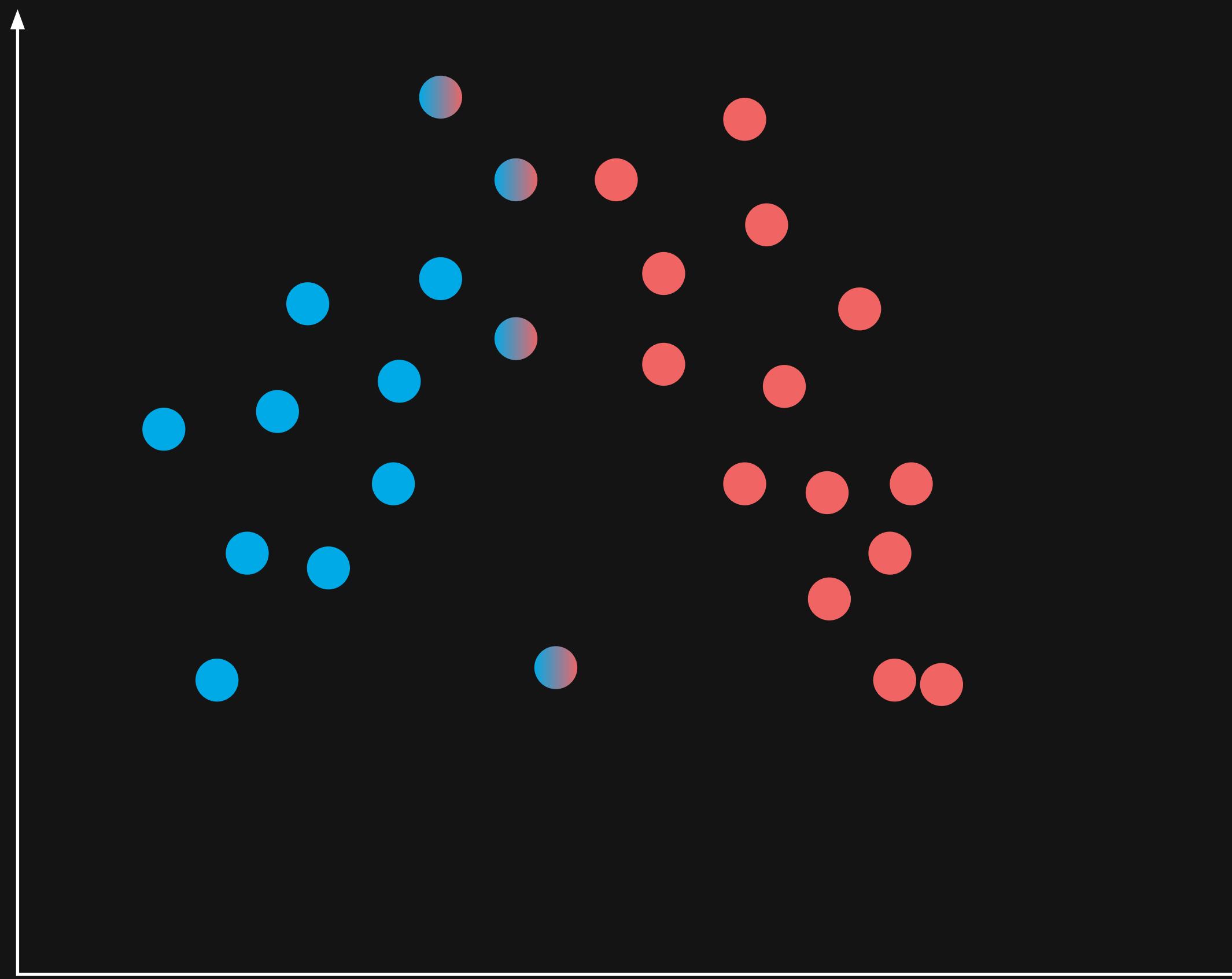


СОДЕРЖАНИЕ

- Обучение без учителя:
кластеризация
- Алгоритмы кластеризации:
 - k-Means
 - EM algorithm
 - DBScan
- Оценка кластеризации



МЯГКАЯ КЛАСТЕРИЗАЦИЯ



ЕМ-АЛГОРИТМ

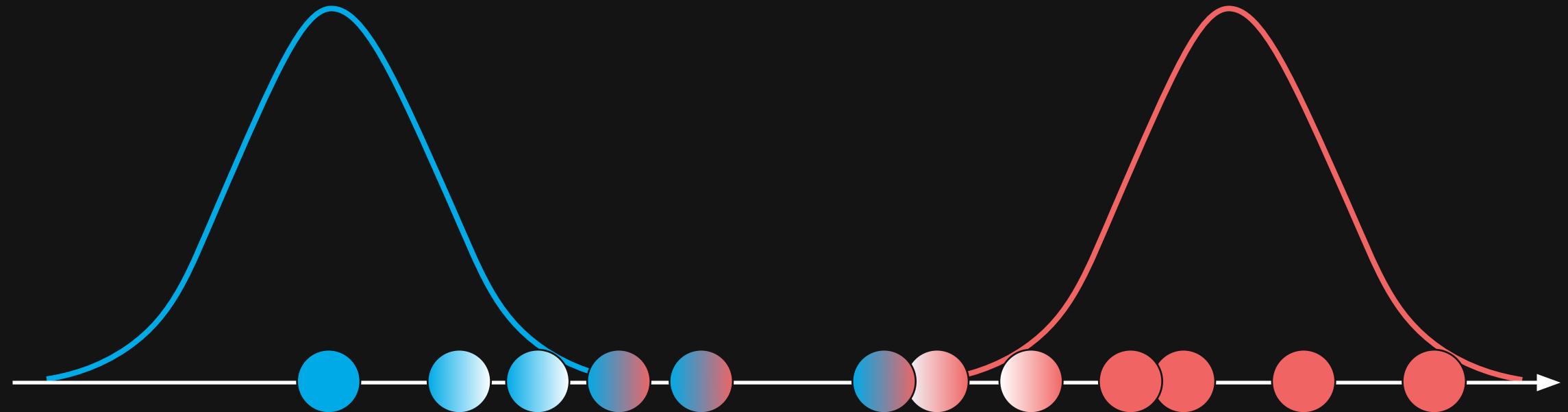
ЕМ (Expectation Maximization) — итеративный алгоритм поиска оценок максимума правдоподобия модели, в ситуации, когда она зависит от скрытых (ненаблюдаемых) переменных

Е-шаг — поиск наиболее вероятных значений скрытых переменных

М-шаг — поиск наиболее вероятных значений параметров для полученных на шаге Е значений скрытых переменных

ЕМ-АЛГОРИТМ (СМЕСЬ ГАУССИАН)

Е-шаг



$$P(x^i|a) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{x^{(i)} - \mu^{(a)}}{2(\sigma^{(i)})^2}\right)$$

$$w_a^{(i)} = \frac{P(x^i|a)P(a)}{P(x^i|a)P(a) + P(x^i|b)P(b)}$$

$$w_b^{(i)} = 1 - w_a^{(i)}$$

$P(x^i|a)$ — вероятность $x(i)$ принадлежать кластеру а

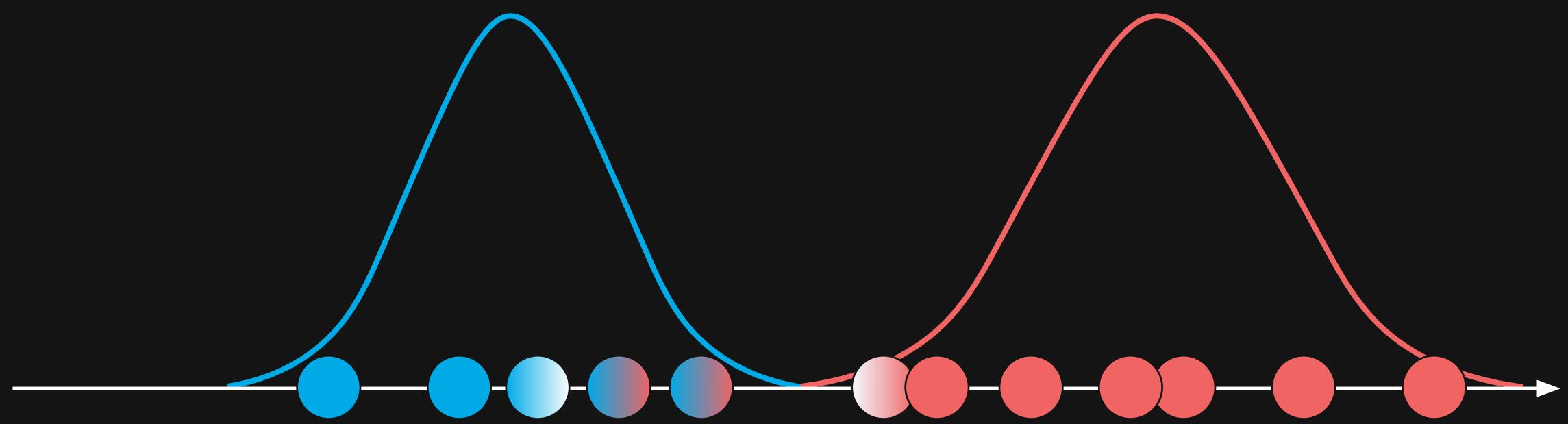
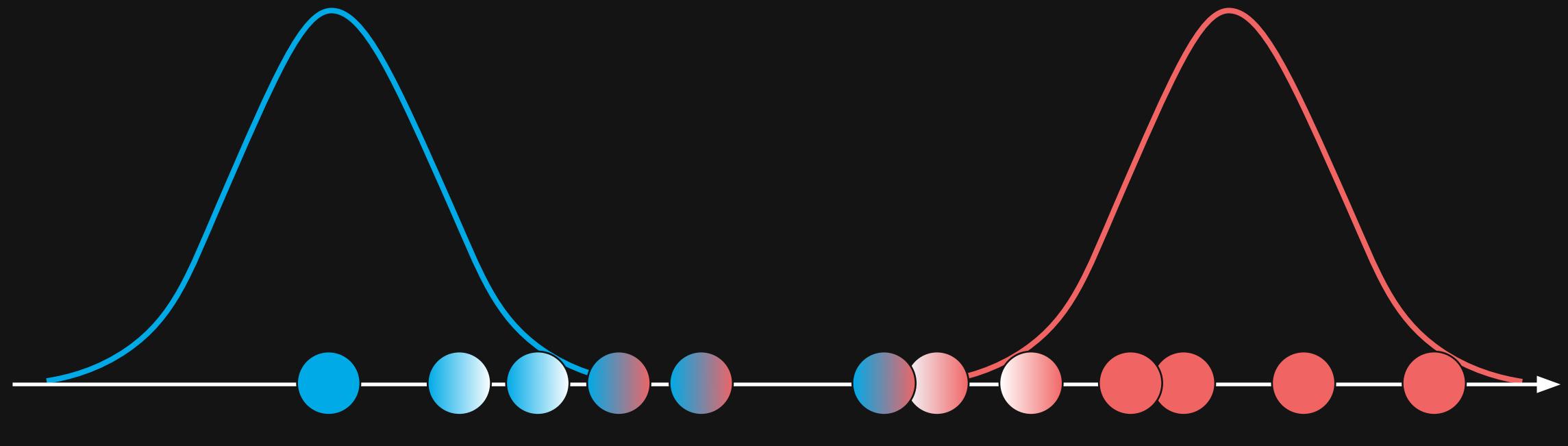
$P(x^i|b)$ — вероятность $x(i)$ принадлежать кластеру б

σ и μ — мат ожидание и стандартное отклонение распределений

w_a и w_b — скрытые переменные

ЕМ-АЛГОРИТМ (СМЕСЬ ГАУССИАН)

М-шаг



$$P(x^i|a) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{x^{(i)} - \mu^{(a)}}{2(\sigma^{(i)})^2}\right)$$

$$P(x^i|b) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(\frac{x^{(i)} - \mu^{(b)}}{2(\sigma^{(i)})^2}\right)$$

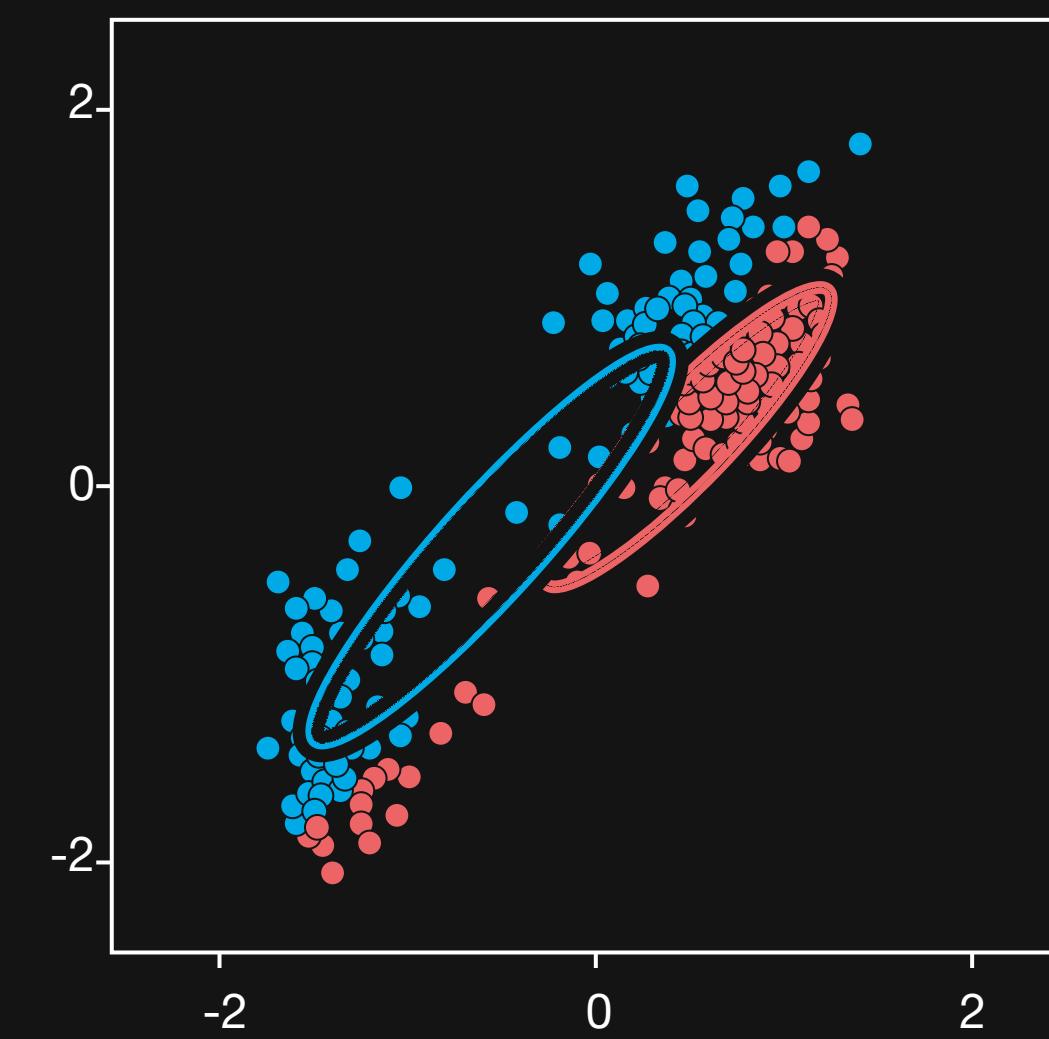
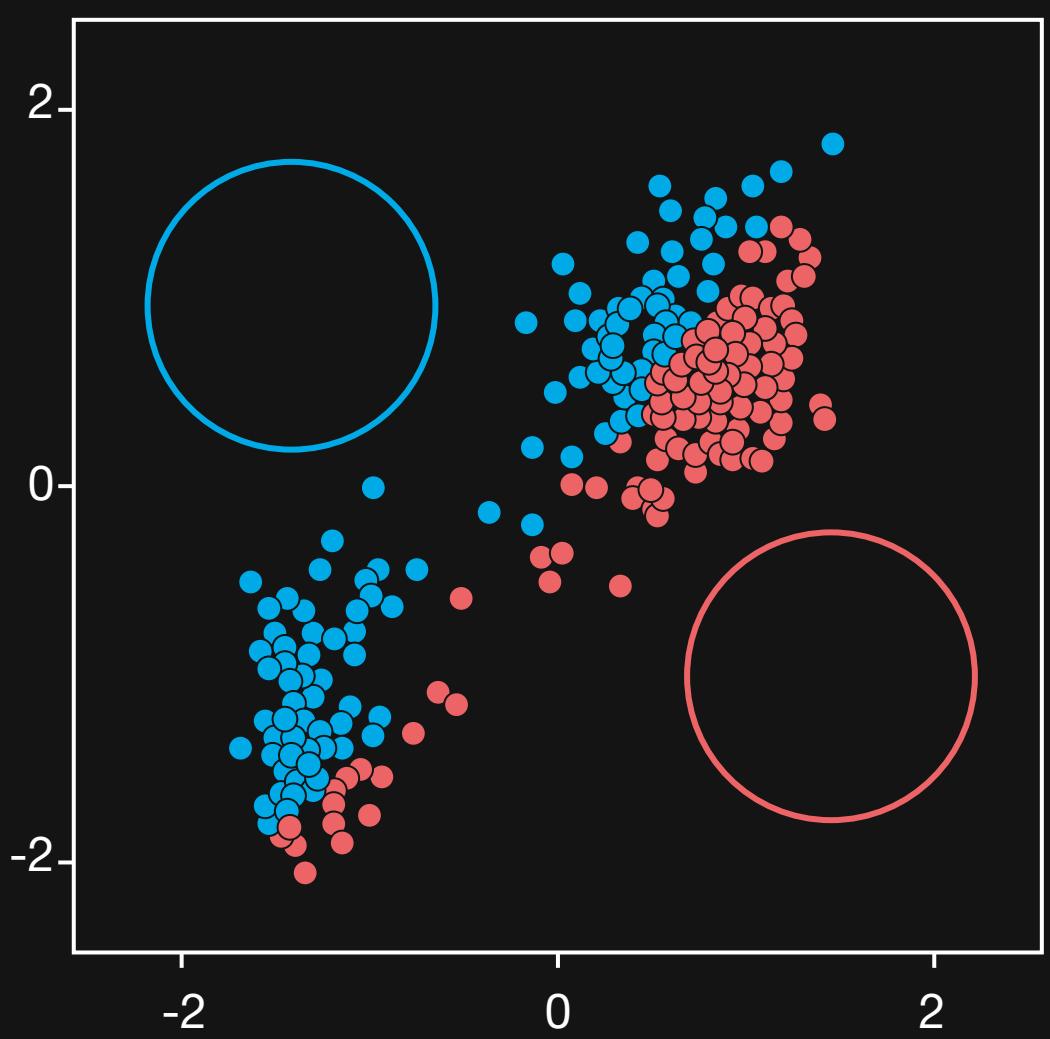
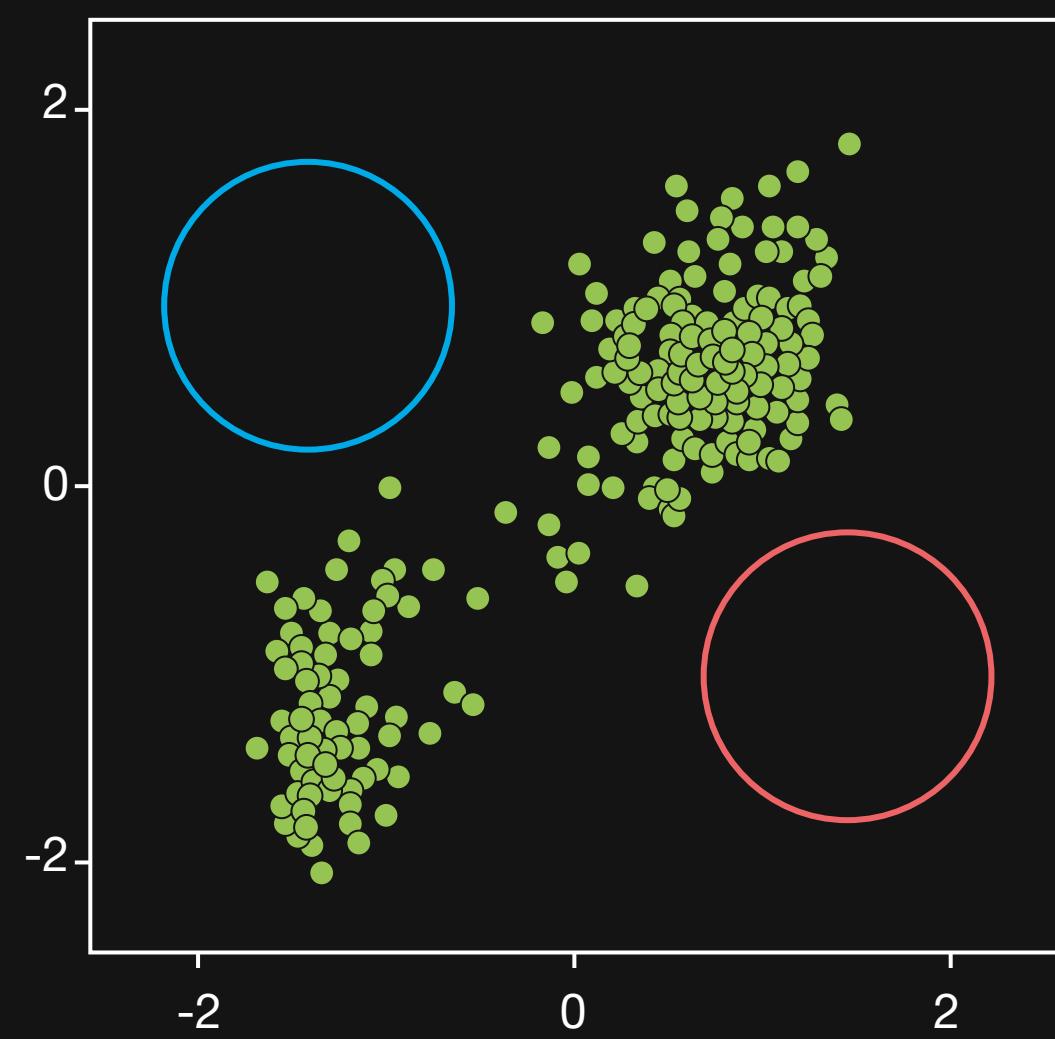
$$w_a^{(i)} = \frac{P(x^i|a)P(a)}{P(x^i|a)P(a) + P(x^i|b)P(b)}$$

$$w_b^{(i)} = 1 - w_a^{(i)}$$

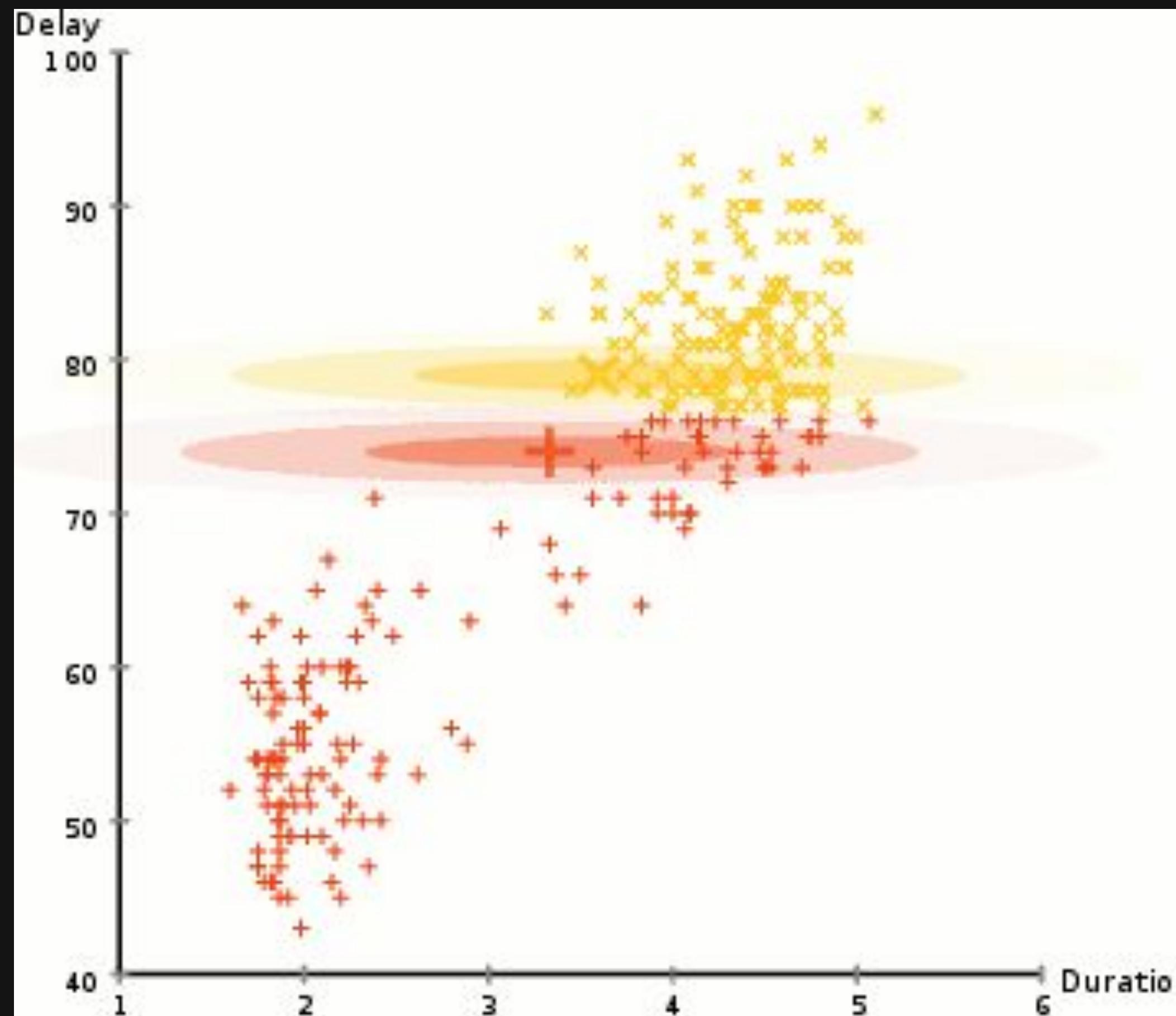
$$\mu_a = \frac{\sum_i w_a^{(i)} x^{(i)}}{\sum_i w^{(i)}}$$

$$\sigma_a = \frac{\sum_i w_a^{(i)} (\mu_a - x^{(i)})^2}{\sum_i w^{(i)}}$$

ЕМ-АЛГОРИТМ (СМЕСЬ ГАУССИАН)



ЕМ-АЛГОРИТМ (СМЕСЬ ГАУССИАН)



ЕМ-АЛГОРИТМ (СМЕСЬ ГАУССИАН)

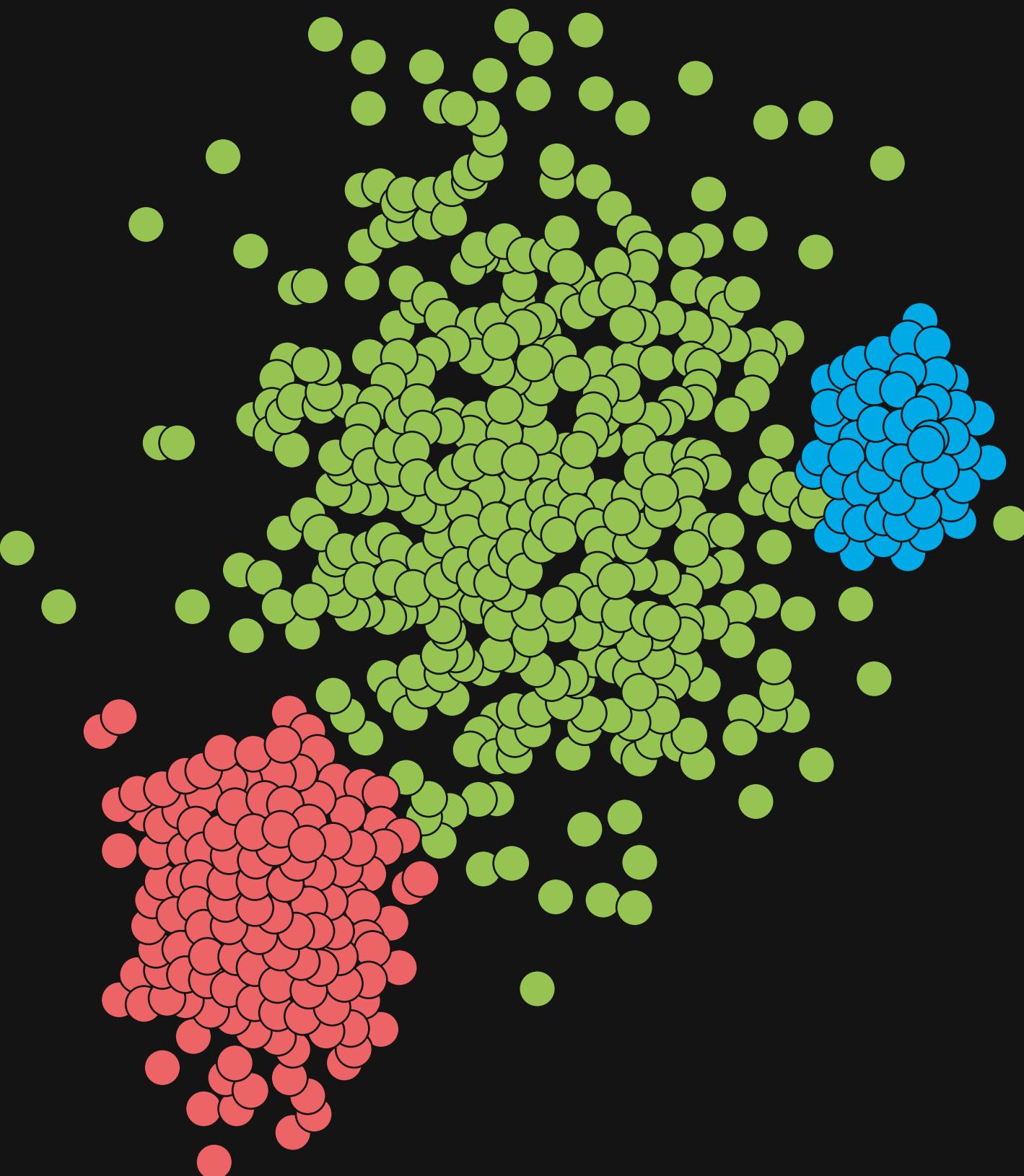
Плюсы:

- Гибкий
- Неплохое качество

Минусы:

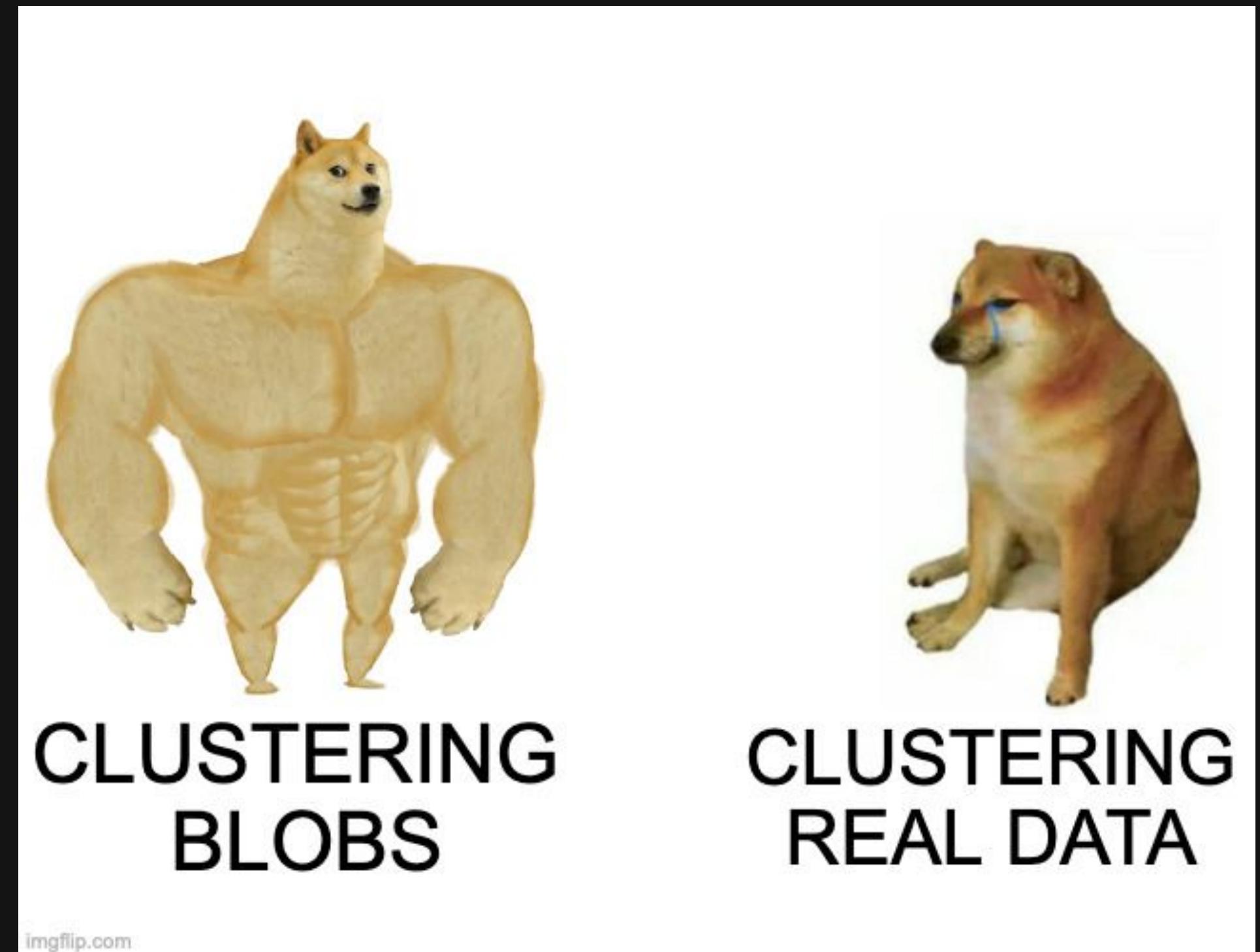
- Нужно выбирать количество кластеров
- Чувствительность к начальным условиям

Кластеры с разной
дисперсией



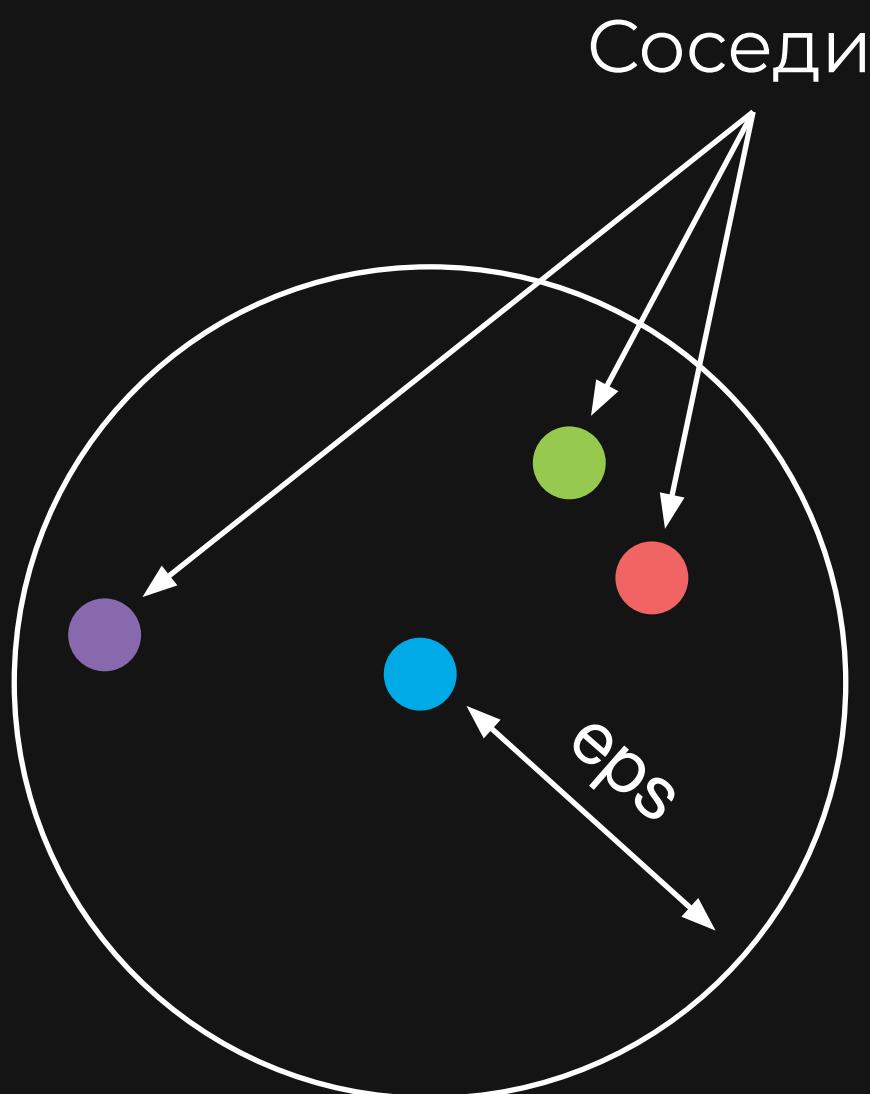
СОДЕРЖАНИЕ

- Обучение без учителя:
кластеризация
- Алгоритмы кластеризации:
 - k-Means
 - EM algorithm
 - DBScan
- Оценка кластеризации



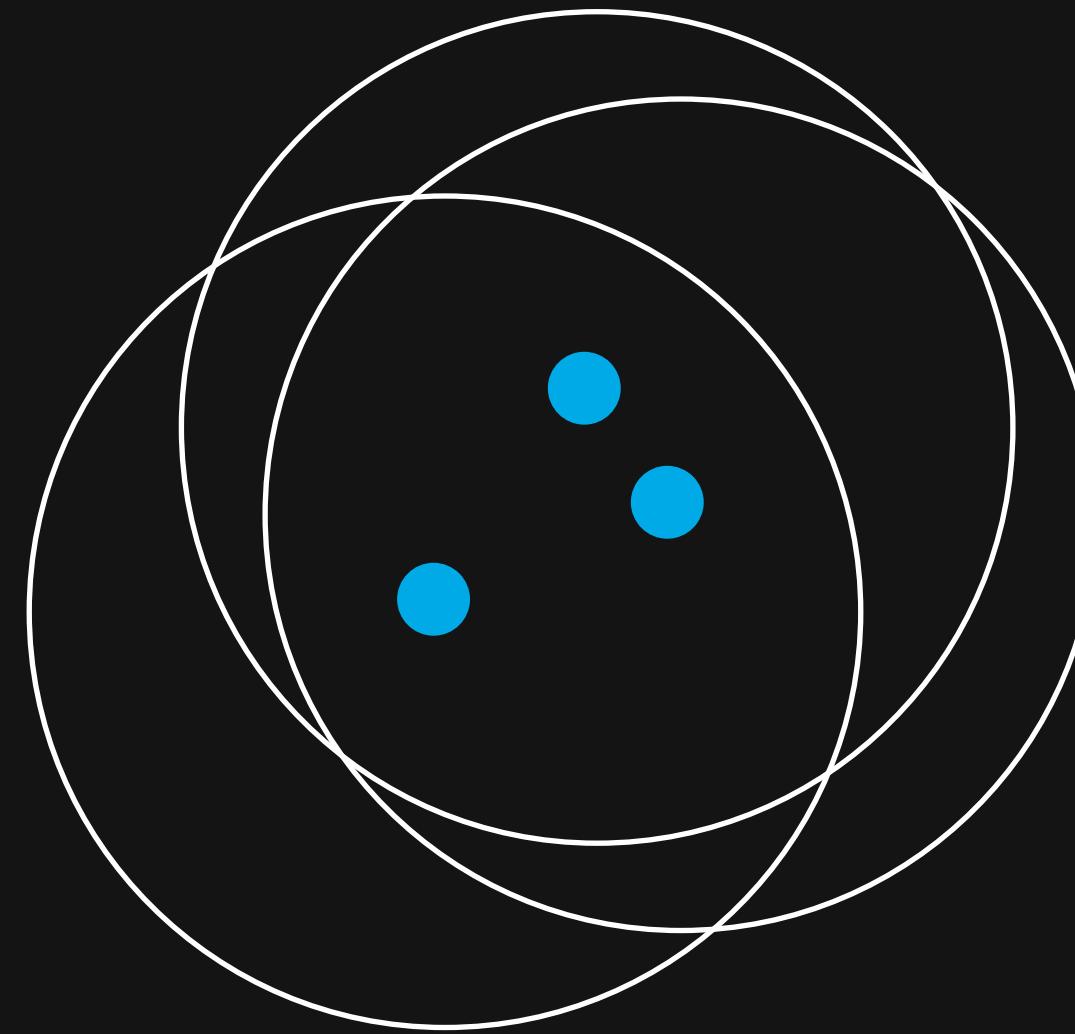
DBSCAN

Density-based spatial clustering of application with noise



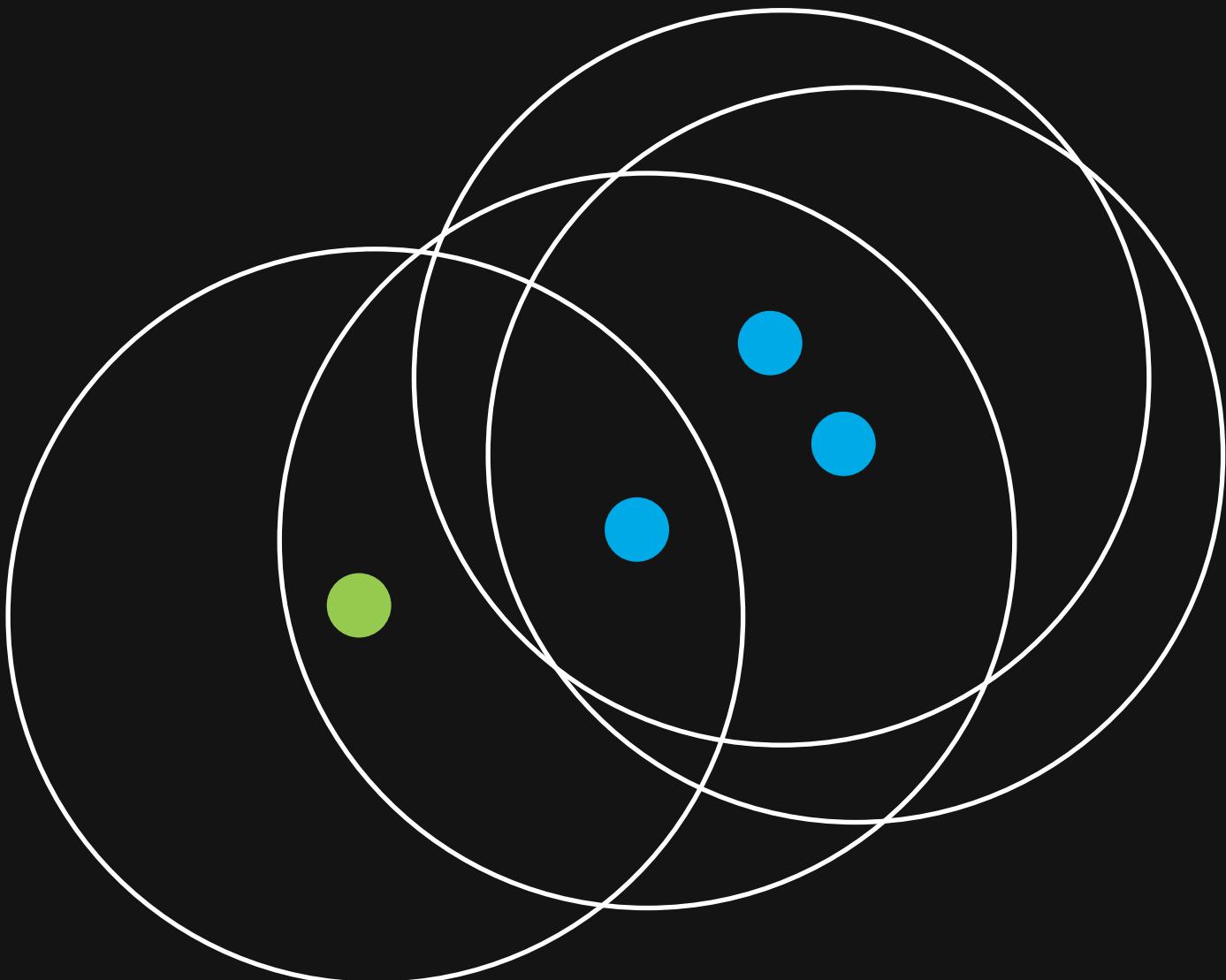
- ▶ eps — радиус, в котором точки считаются соседями
- ▶ min_samples — минимальное число соседей, чтобы считать точку внутрикластерной

DBSCAN



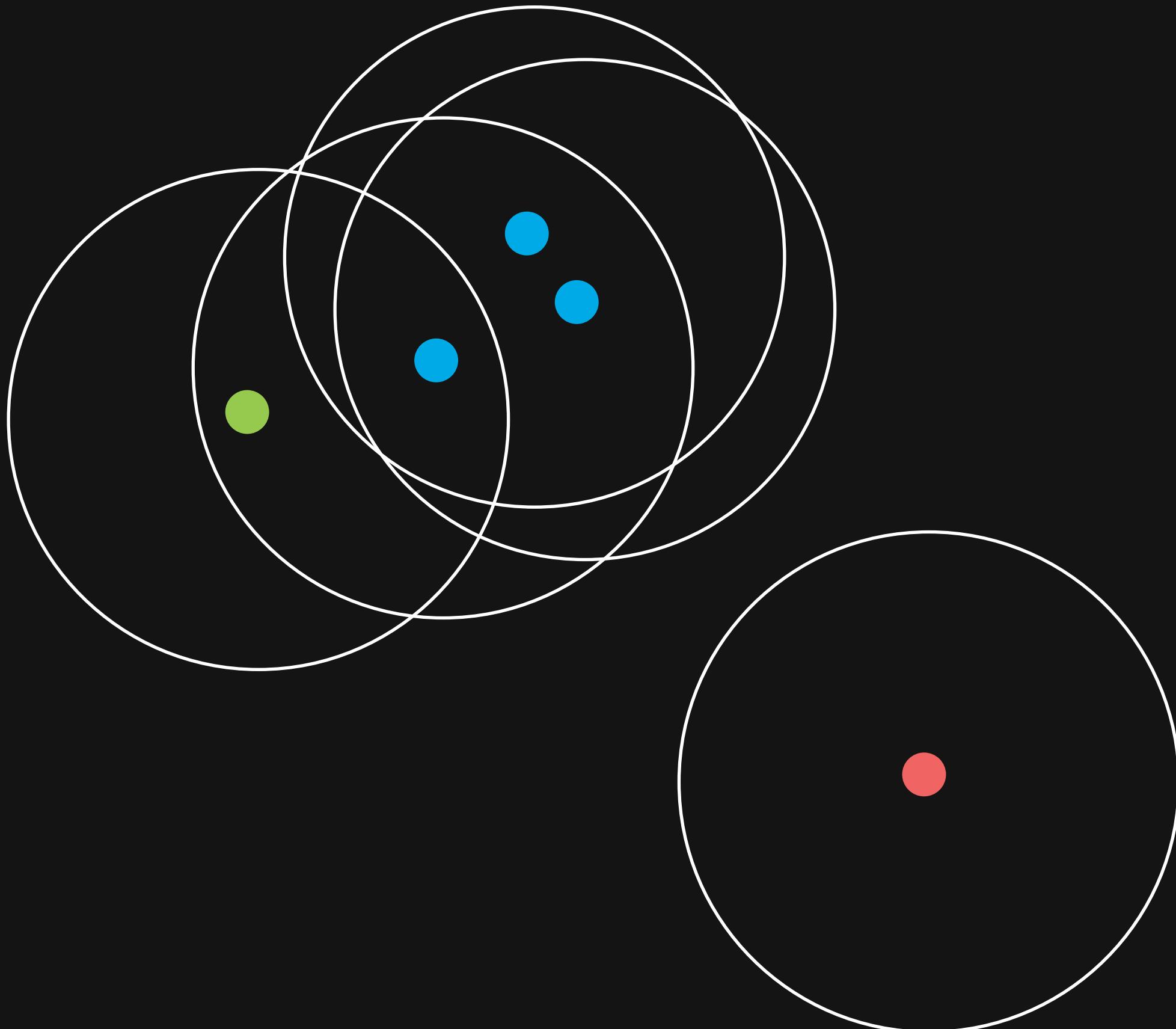
- ▶ `eps`
- ▶ `min_samples = 2`
- ▶ Точка называется **основной**,
если в радиусе `eps` как
минимум `min_samples`
соседей

DBSCAN



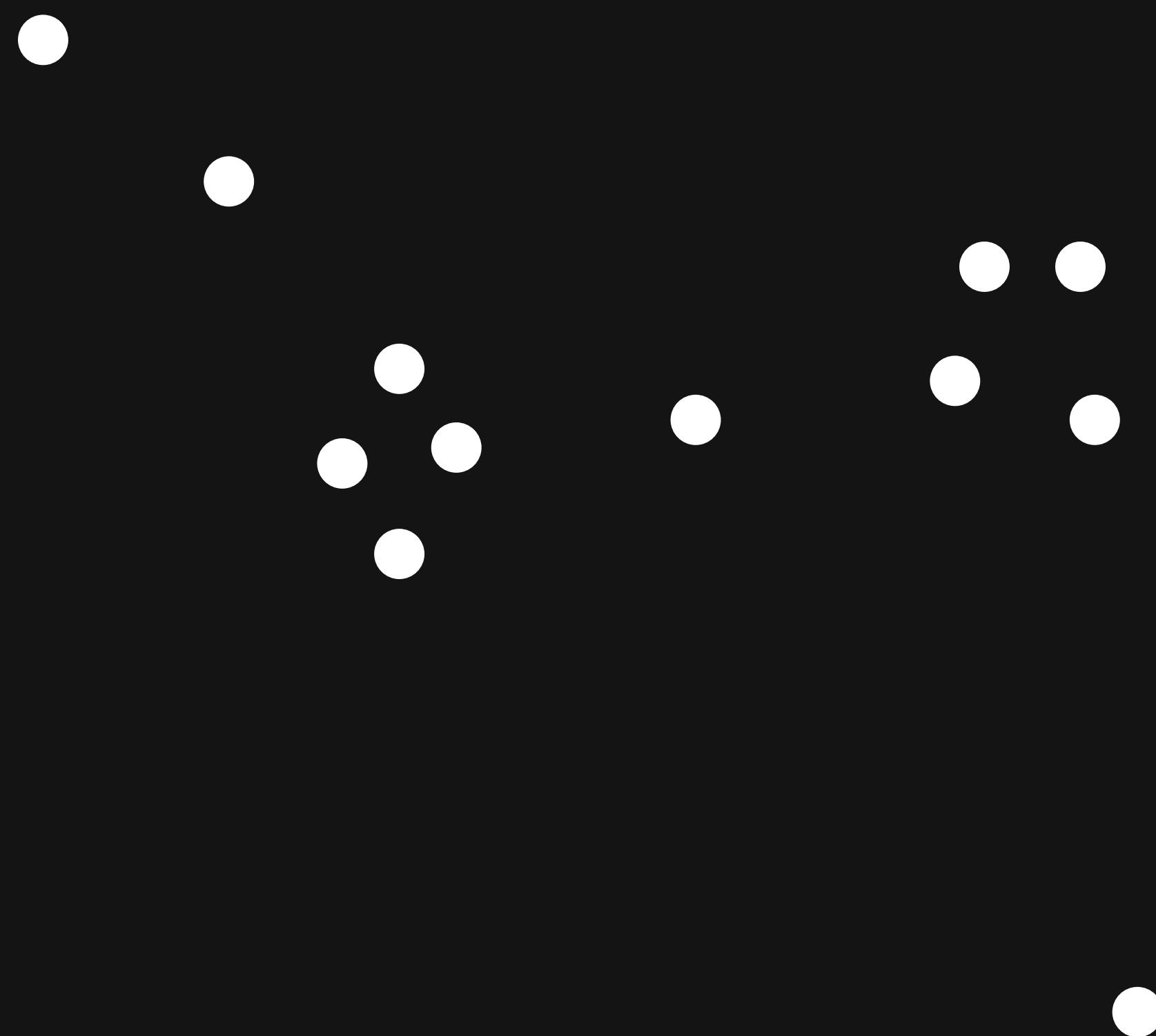
- ▶ eps
- ▶ $\text{min_samples} = 2$
- ▶ Точка называется **основной**, если в радиусе eps как минимум min_samples соседей
- ▶ Точка называется **краевой**, если она попадает в радиус основной, но в радиусе краевой нет min_samples точек

DBSCAN



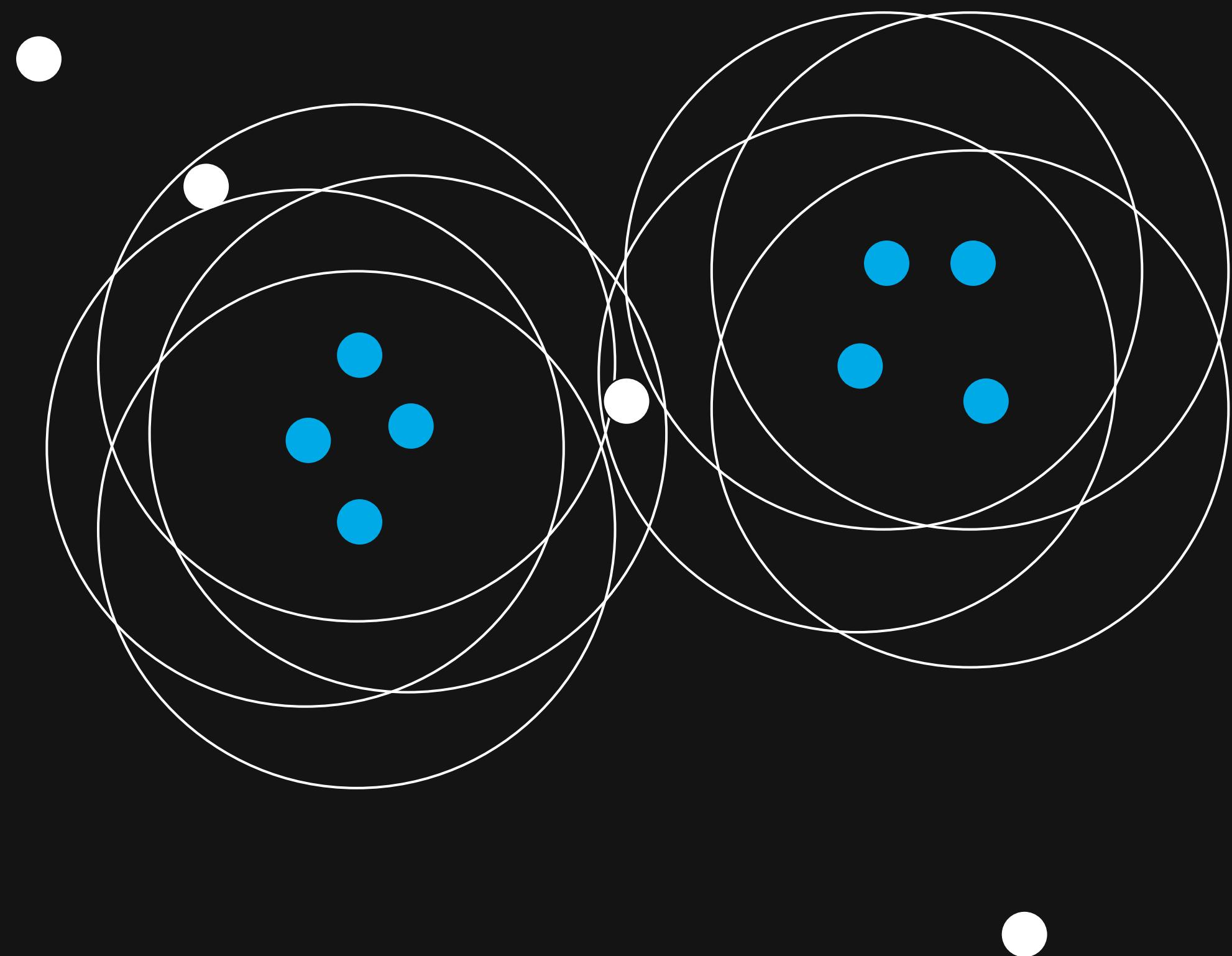
- ▶ `eps`
- ▶ `min_samples = 2`
- ▶ Точка называется **основной**, если в радиусе `eps` как минимум `min_samples` соседей
- ▶ Точка называется **краевой**, если она попадает в радиус основной, но в радиусе краевой нет `min_samples` точек
- ▶ Точка называется **выбросом**, если в ее радиусе `eps` нет внутрикластерной точек и меньше `min_samples` точек

DBSCAN



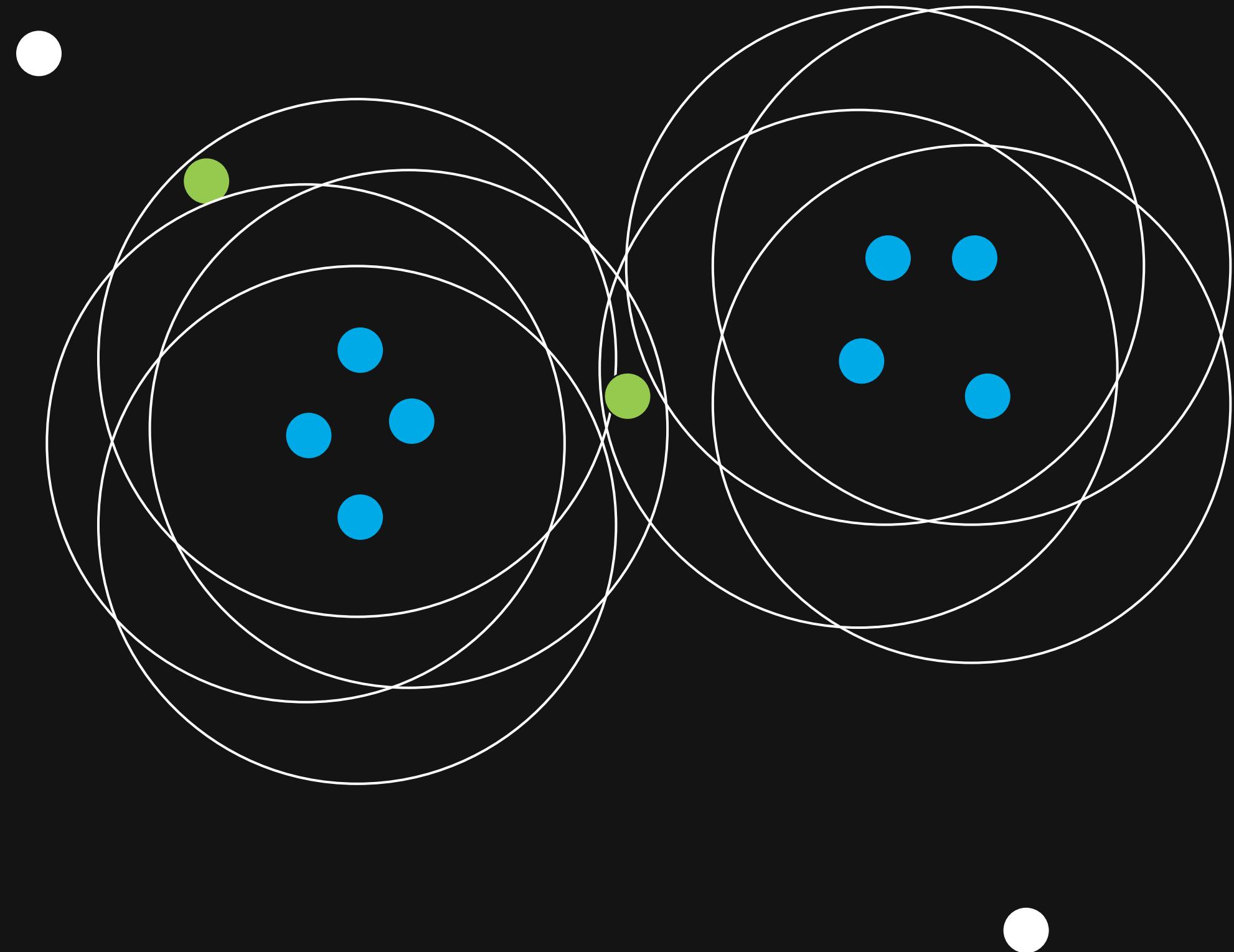
- ▶ `eps`
- ▶ `min_samples = 3`

DBSCAN



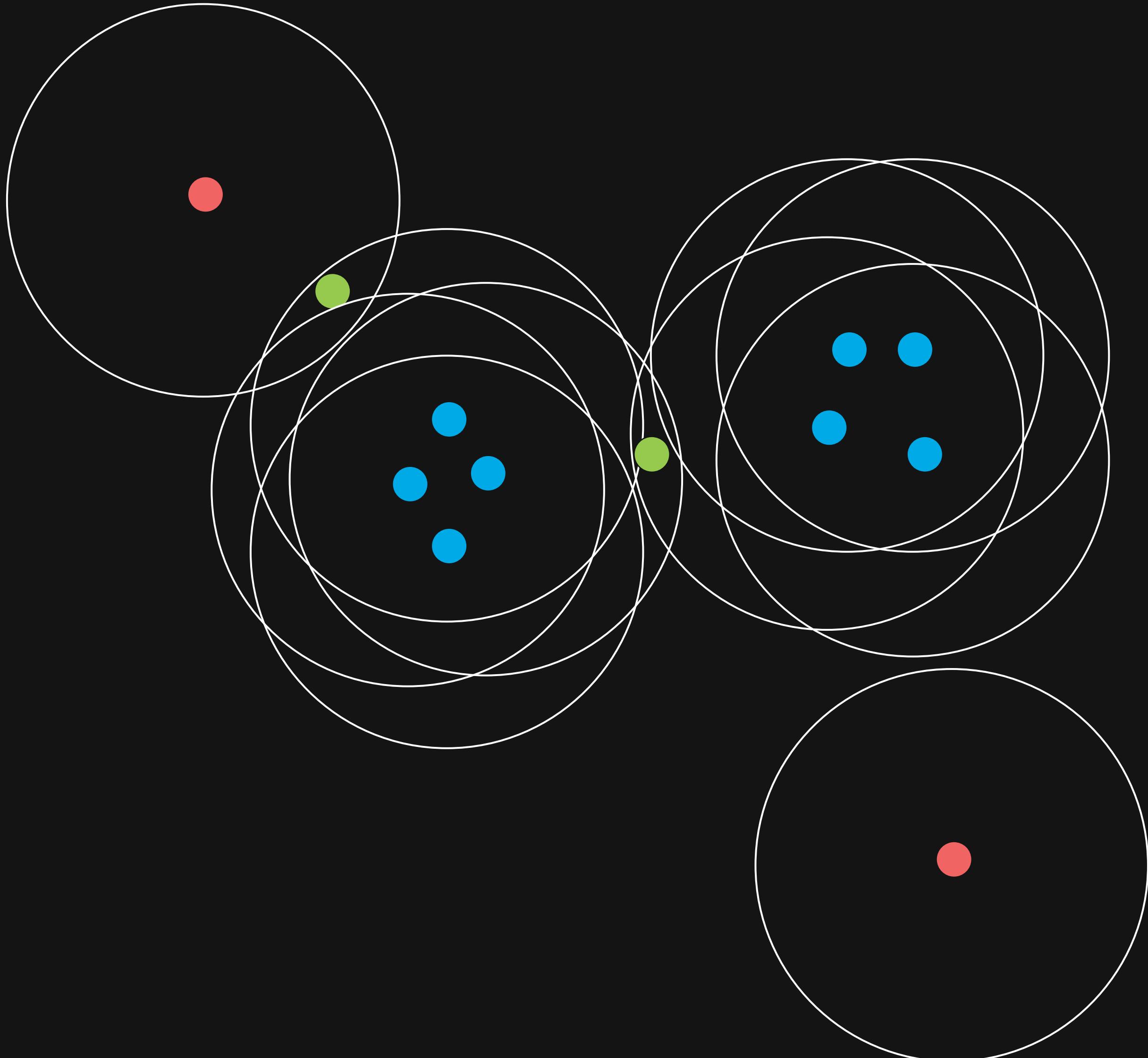
- ▶ `eps`
- ▶ `min_samples = 3`
- ▶ Для каждой точки считаем число ее соседей. Определяем **основные** точки. Основная точка и достижимые из нее объединяются в кластер

DBSCAN



- ▶ eps
- ▶ $\text{min_samples} = 3$
- ▶ Для каждой точки считаем число ее соседей. Определяем **основные** точки. Основная точка и достижимые из нее объединяются в кластер
- ▶ Определяем **краевые** точки, т.е. такие, в радиусе eps которых есть основные.

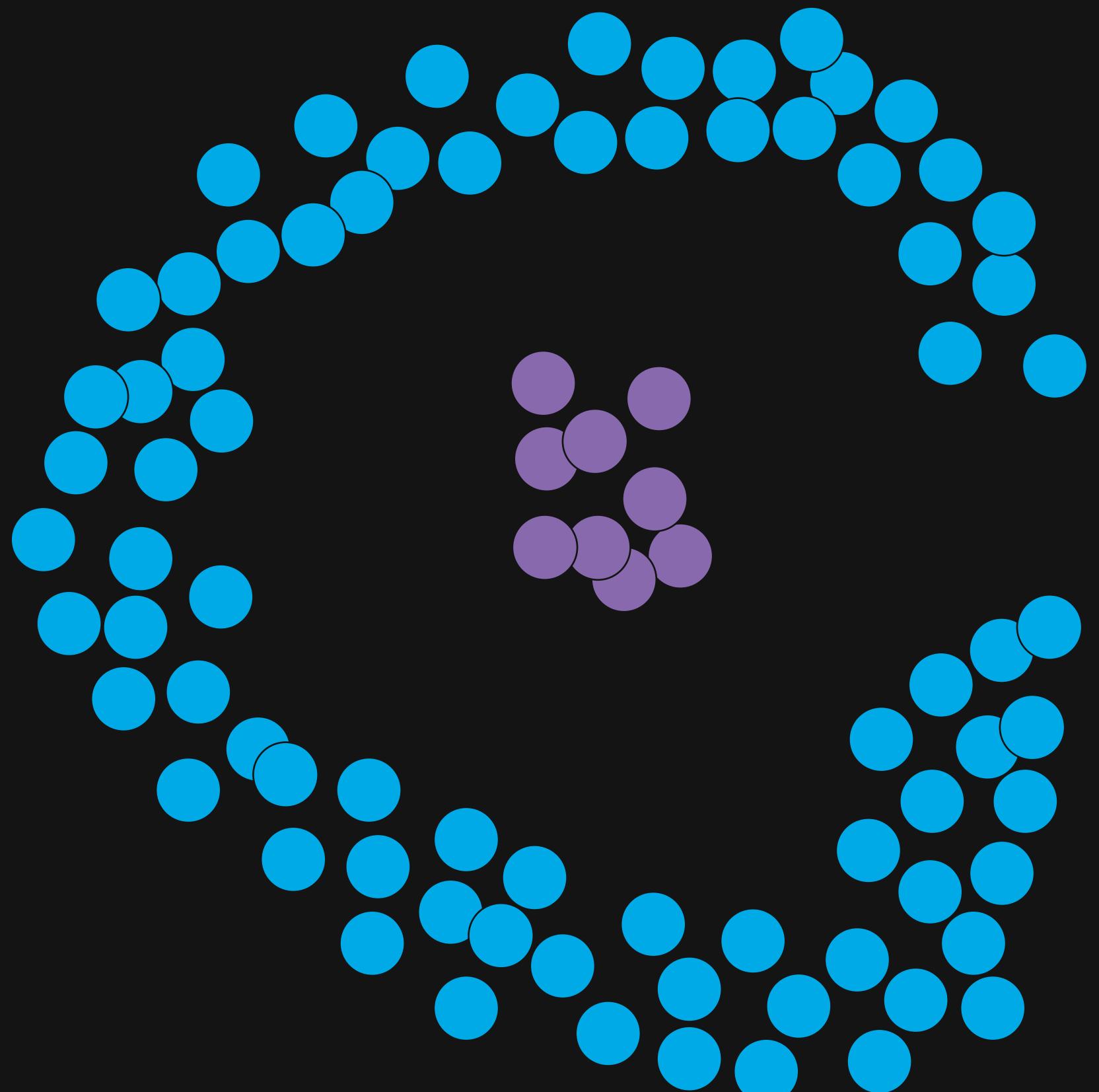
DBSCAN



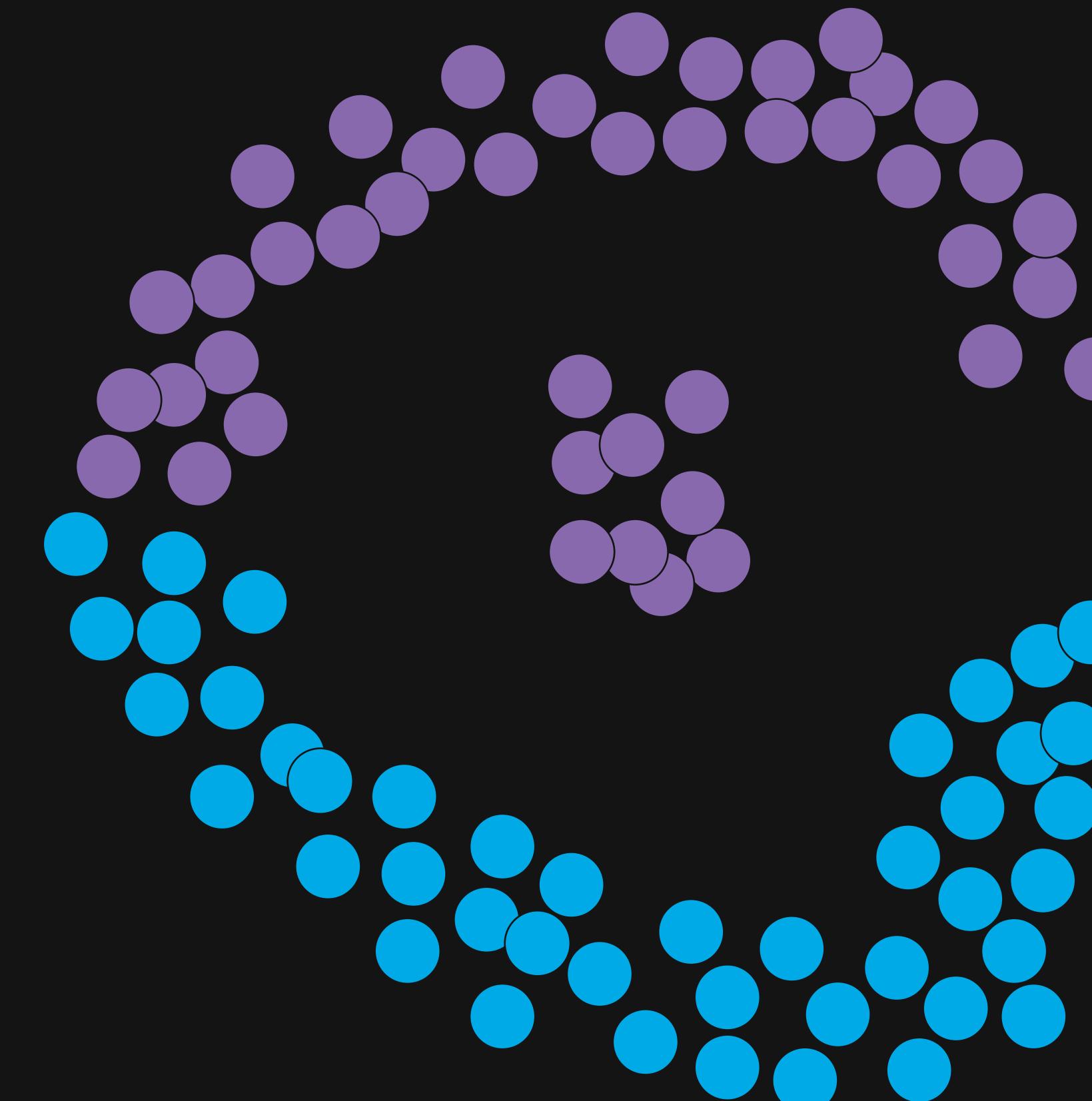
- ▶ eps
- ▶ $\text{min_samples} = 3$
- ▶ Для каждой точки считаем число ее соседей. Определяем **основные** точки. Основная точка и достижимые из нее объединяются в кластер
- ▶ Определяем **краевые** точки, т. е. такие, в радиусе eps которых есть основные.
- ▶ Остальные точки — это **выбросы**

DBSCAN

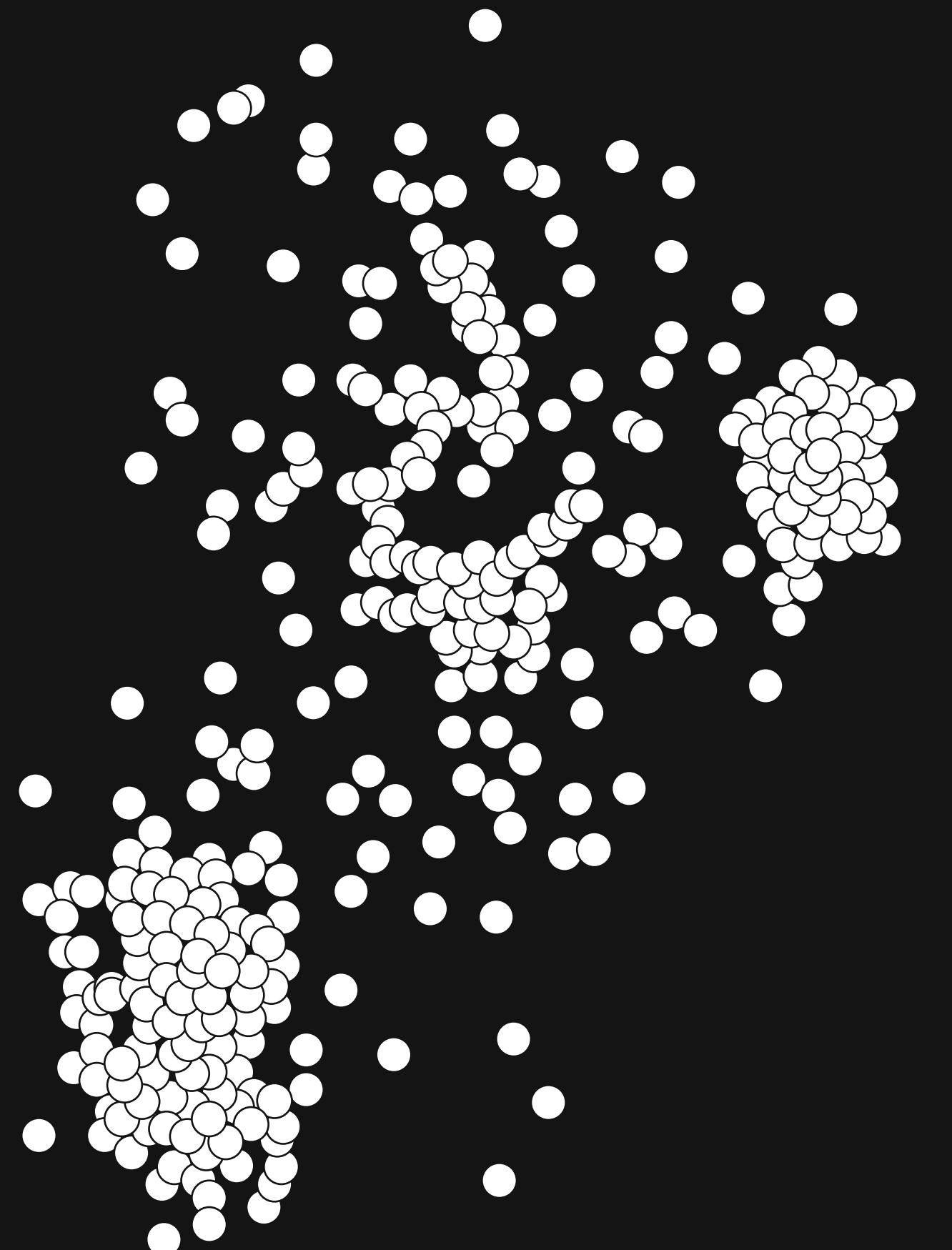
DBScan



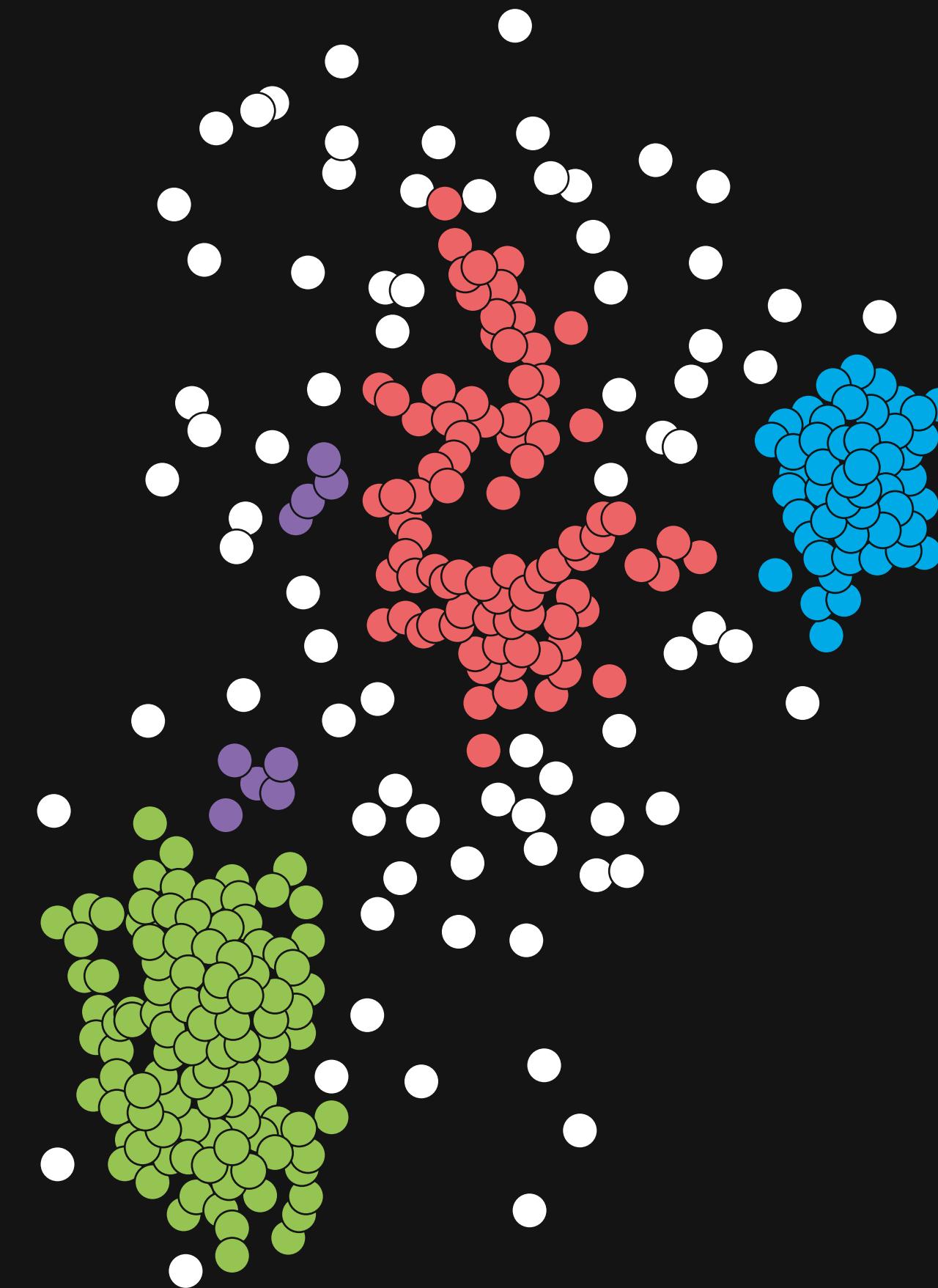
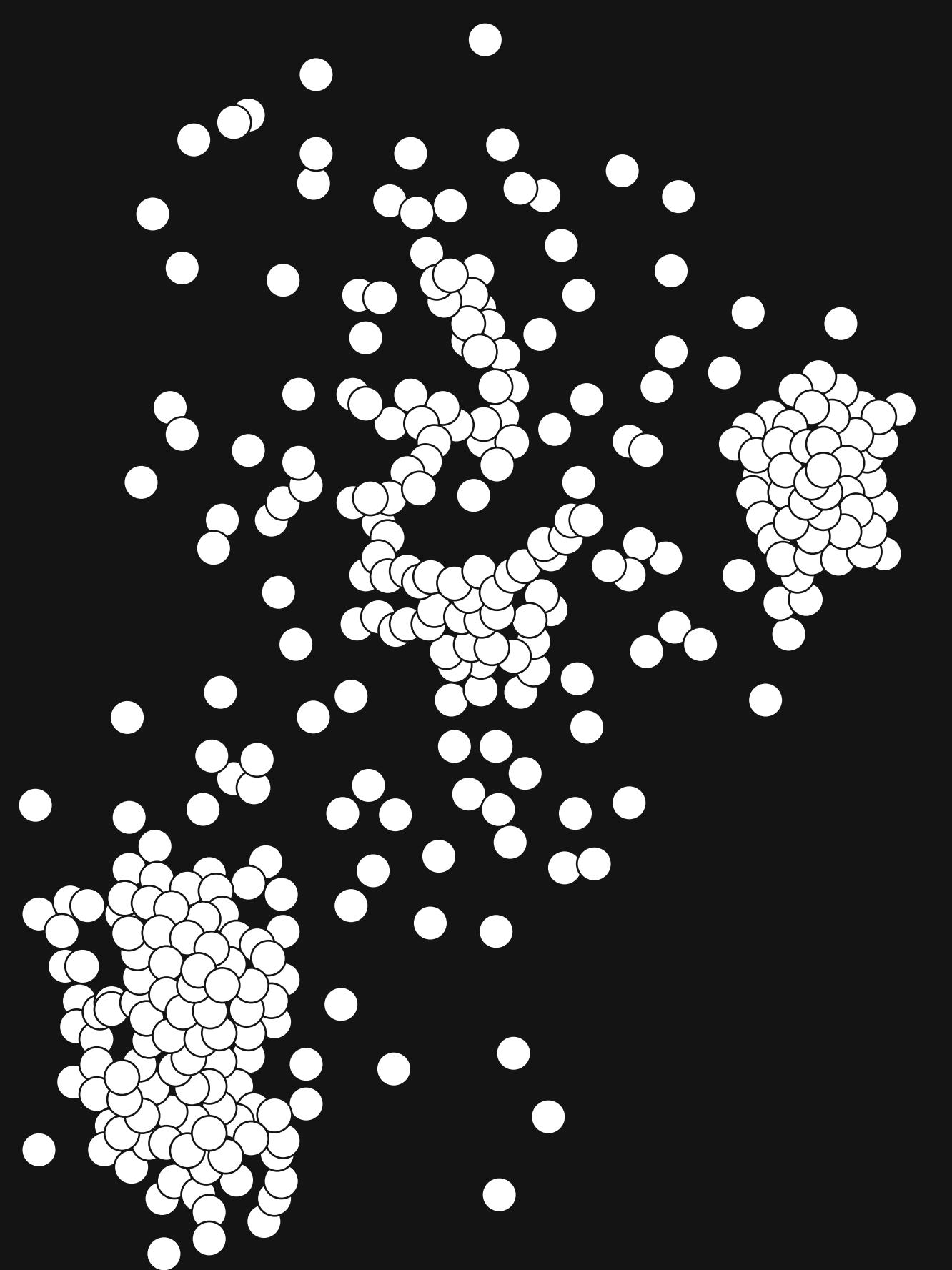
K-means



DBSCAN



DBSCAN



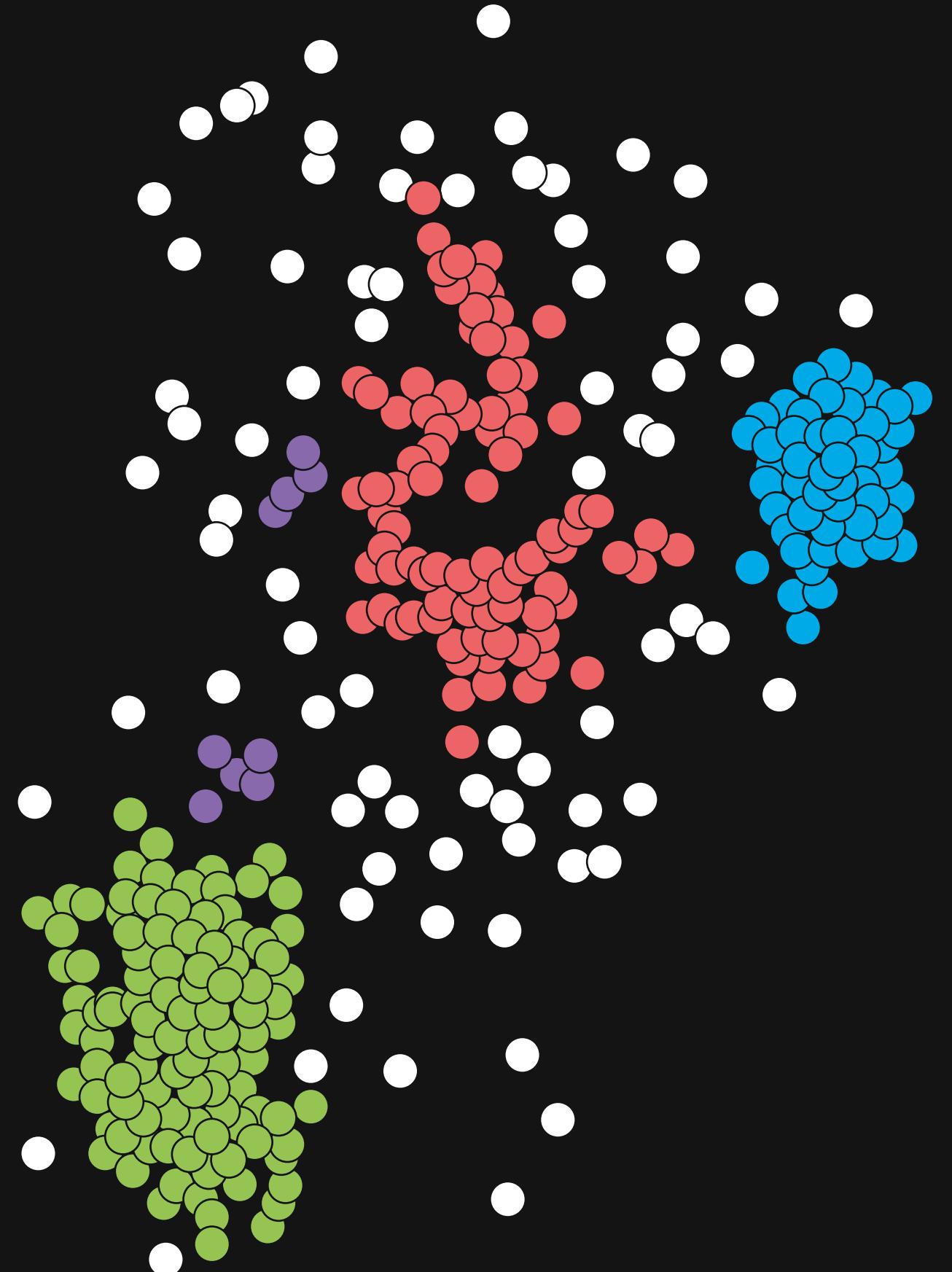
DBSCAN

Плюсы:

- Не требует спецификации числа кластеров
- Кластеры произвольной формы
- Устойчив к выбросам

Минусы:

- Зависит от измерения расстояния
- Не может хорошо кластеризовать наборы данных с большой разницей в плотности

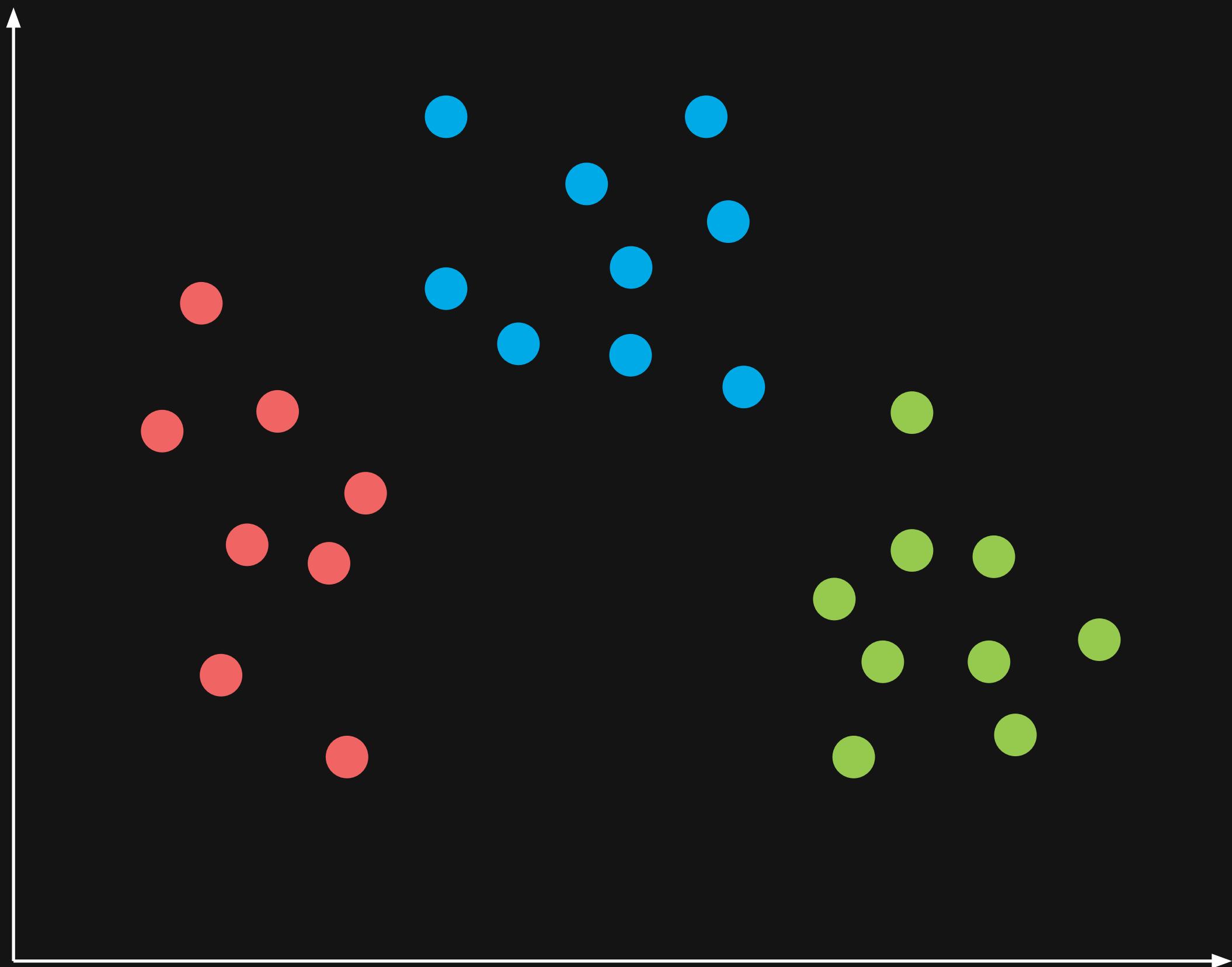


СОДЕРЖАНИЕ

- Обучение без учителя:
кластеризация
- Алгоритмы кластеризации:
 - k-Means
 - EM algorithm
 - DBScan
- Оценка кластеризации

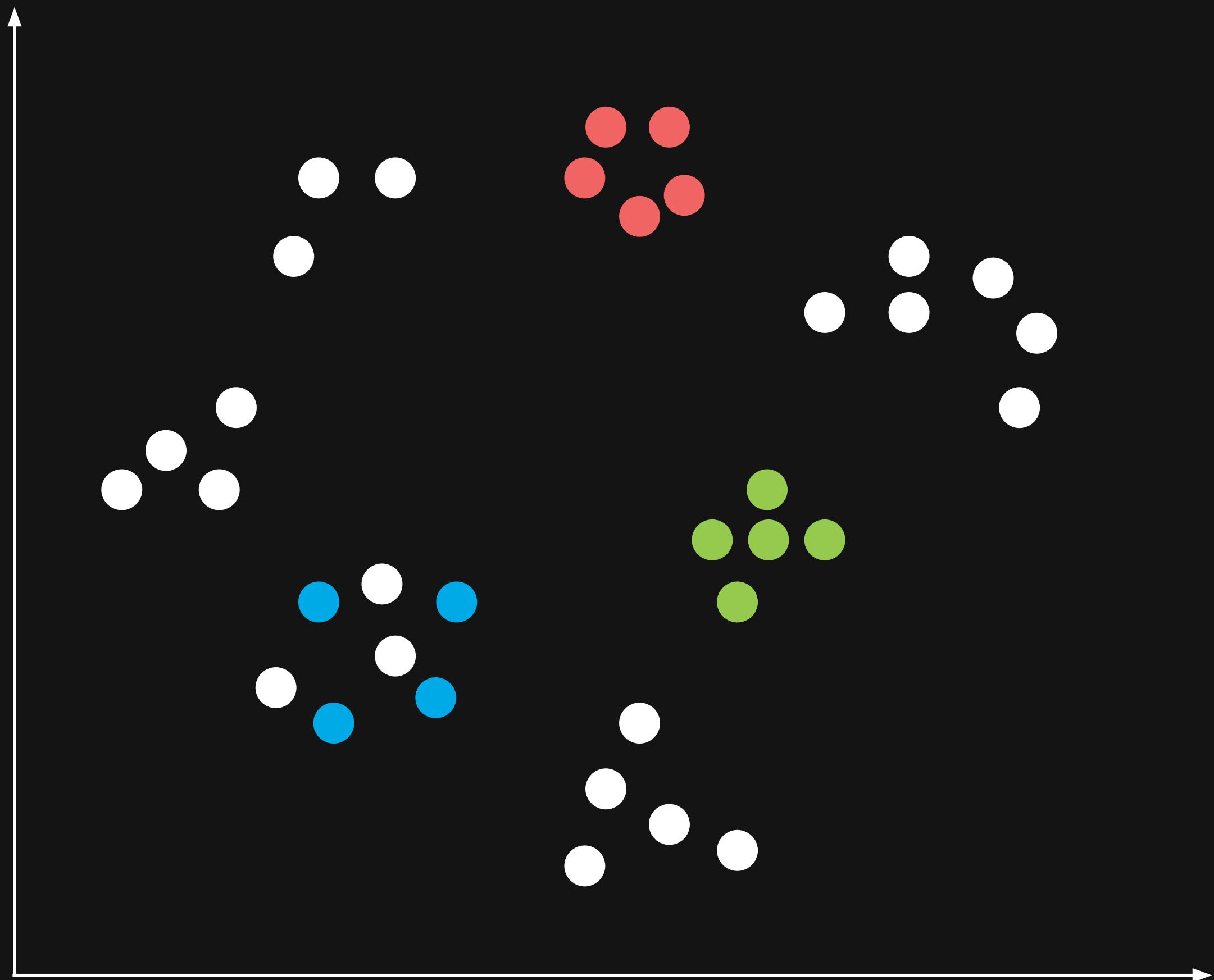


ОЦЕНКА КЛАСТЕРИЗАЦИИ



- Внешние меры оценки качества
- Внутренние меры оценки качества

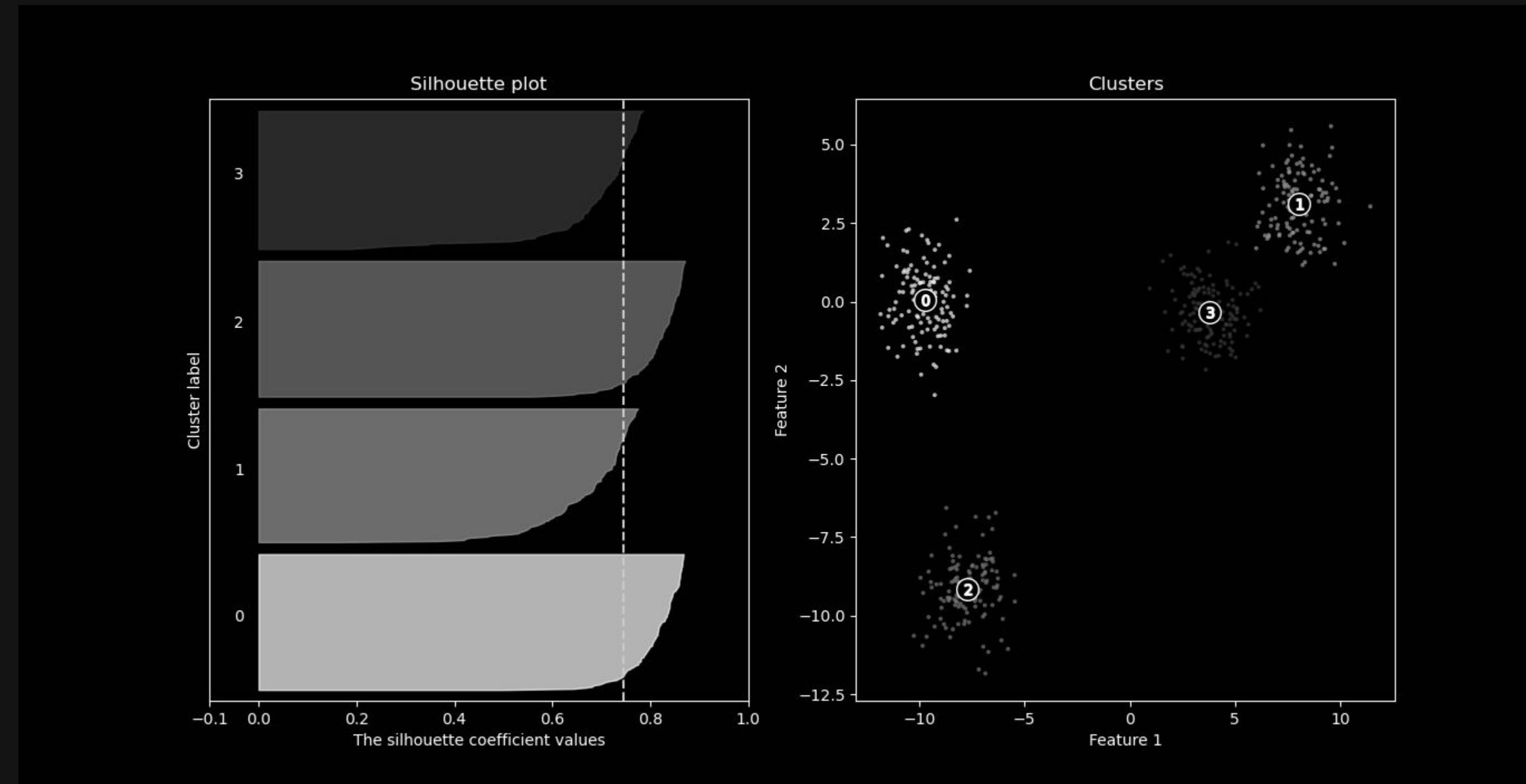
ВНЕШНИЕ МЕРЫ ОЦЕНКИ КАЧЕСТВА: ИНДЕКС RAND



- TP - Элементы принадлежат одному кластеру и одному классу
- FP - Элементы принадлежат одному кластеру, но разным классам
- FN - Элементы принадлежат разным кластерам, но одному классу
- TN - Элементы принадлежат разным кластерам и разным классам

$$Rand = \frac{TP + TN}{TP + TN + FP + FN}$$

ВНУТРЕННИЕ МЕРЫ ОЦЕНКИ КАЧЕСТВА: SILHOUETTE

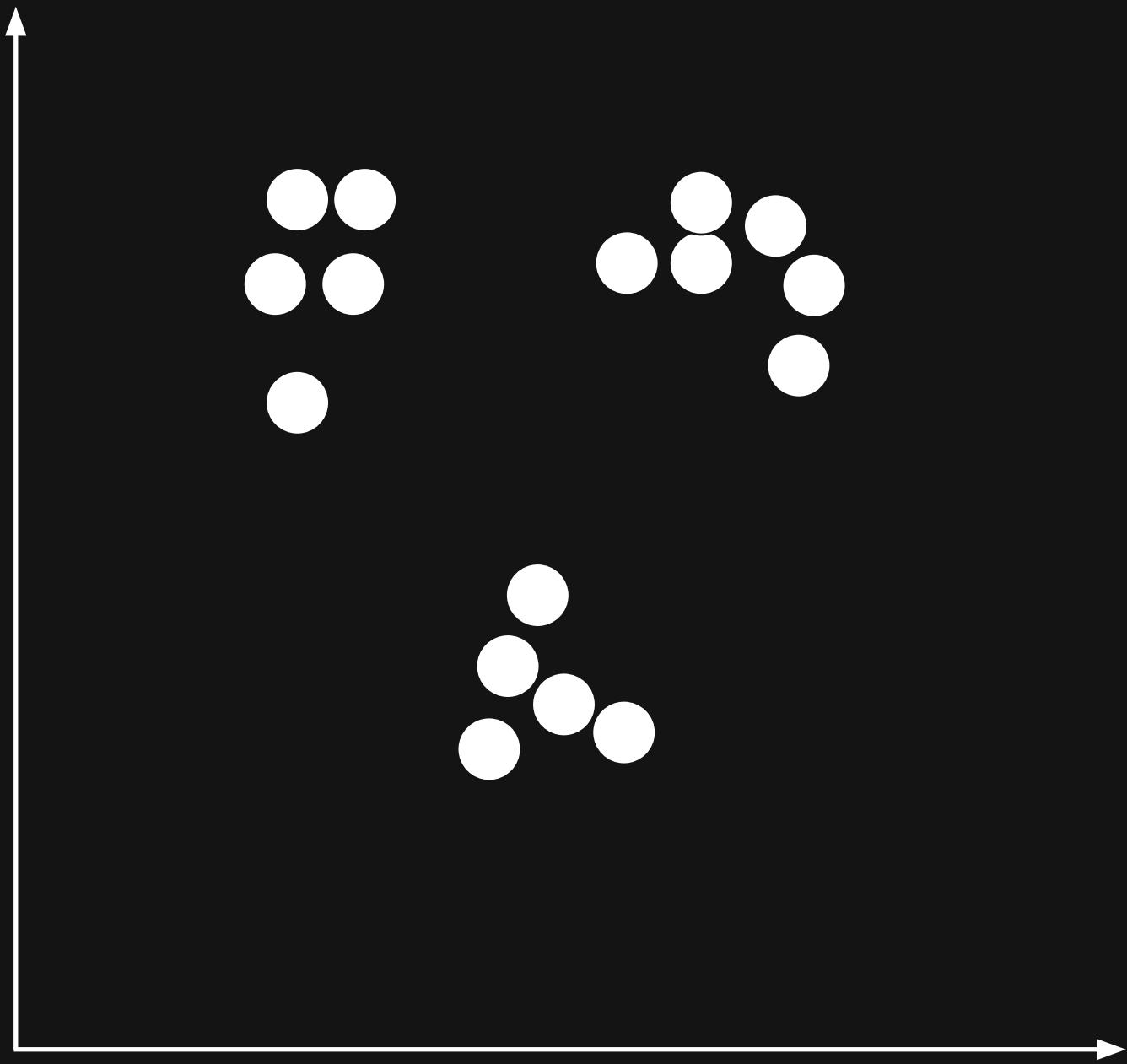


$$silhouette = \frac{(b - a)}{max(a, b)}$$

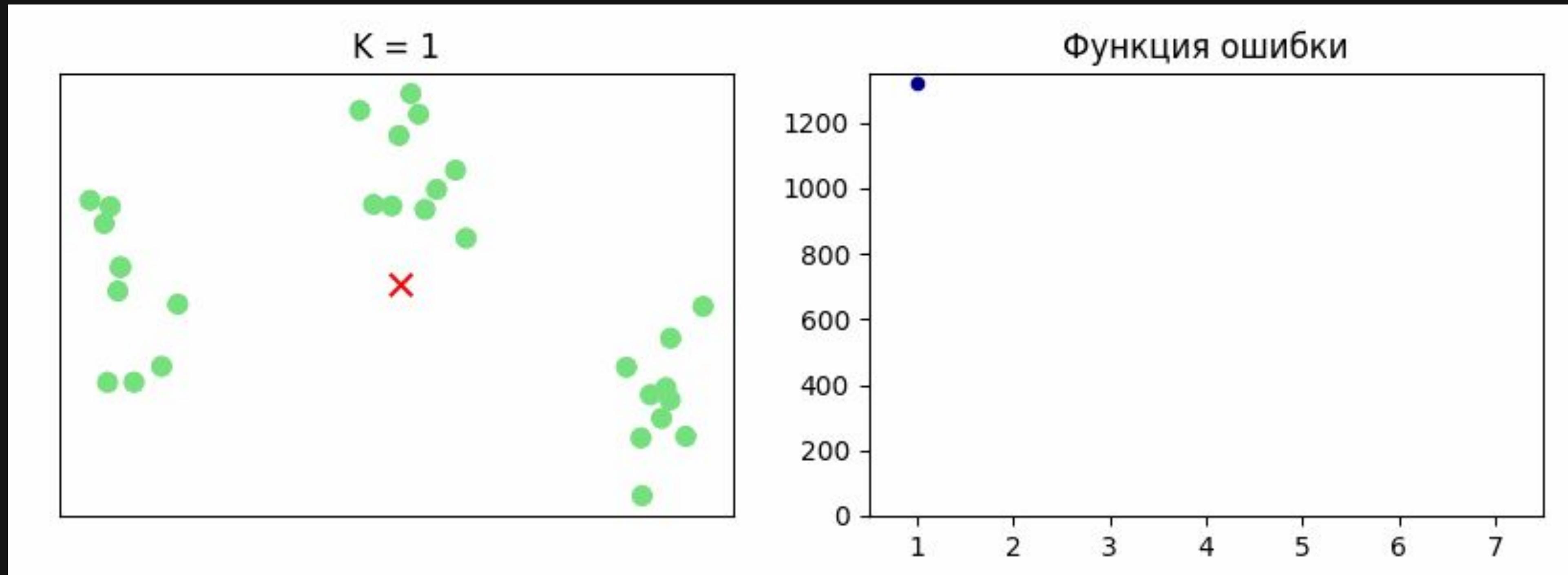
а — среднее внутрикластерное
расстояние
б — среднее расстояние до
ближайшего кластера

КОЛИЧЕСТВО КЛАСТЕРОВ: ПРАВИЛО ЛОКТА

$$V = \sum_{i=1}^k \sum_{x \in S_i} (x - \mu_i)^2$$

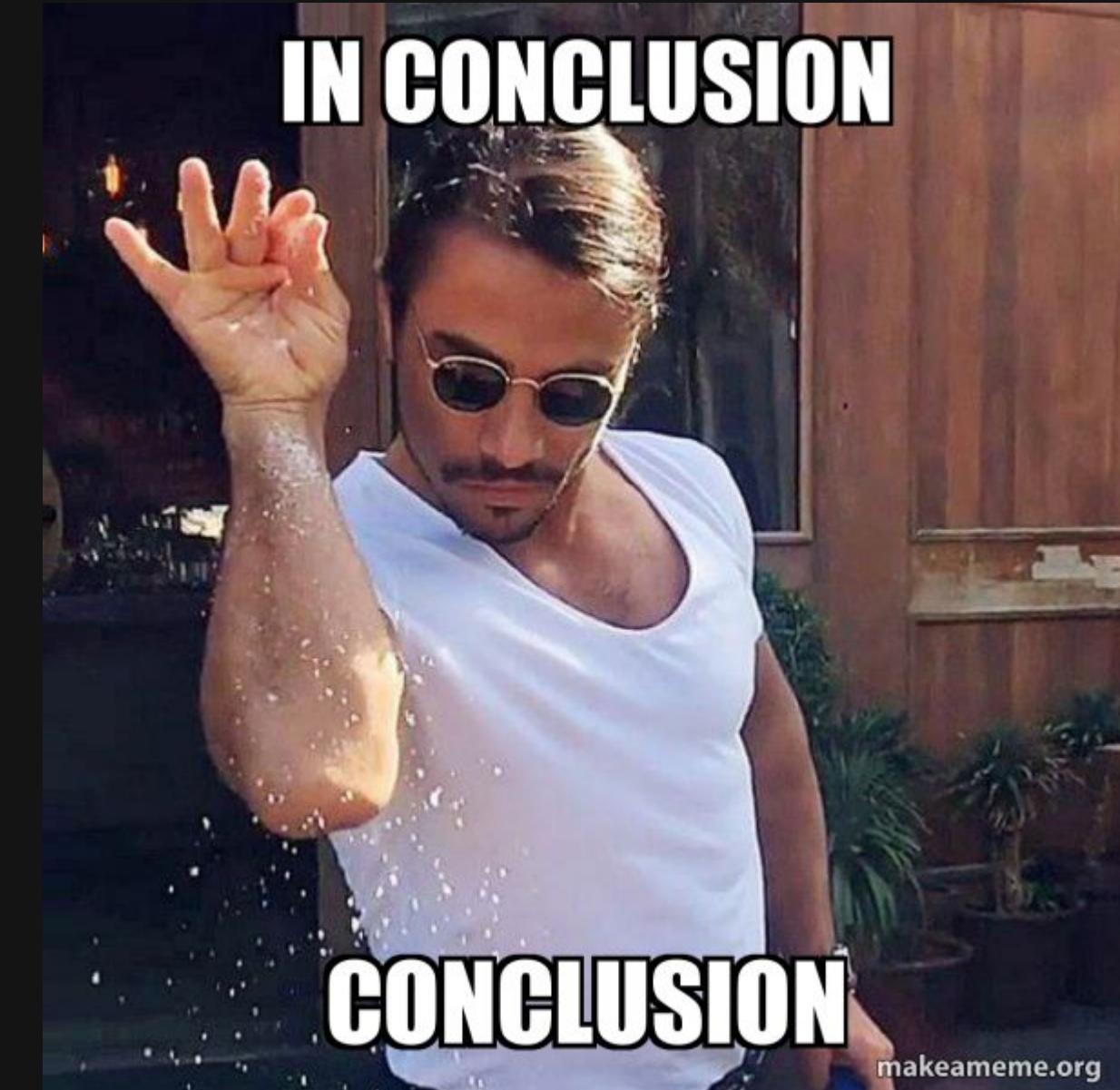


КОЛИЧЕСТВО КЛАСТЕРОВ: ПРАВИЛО ЛОКТА

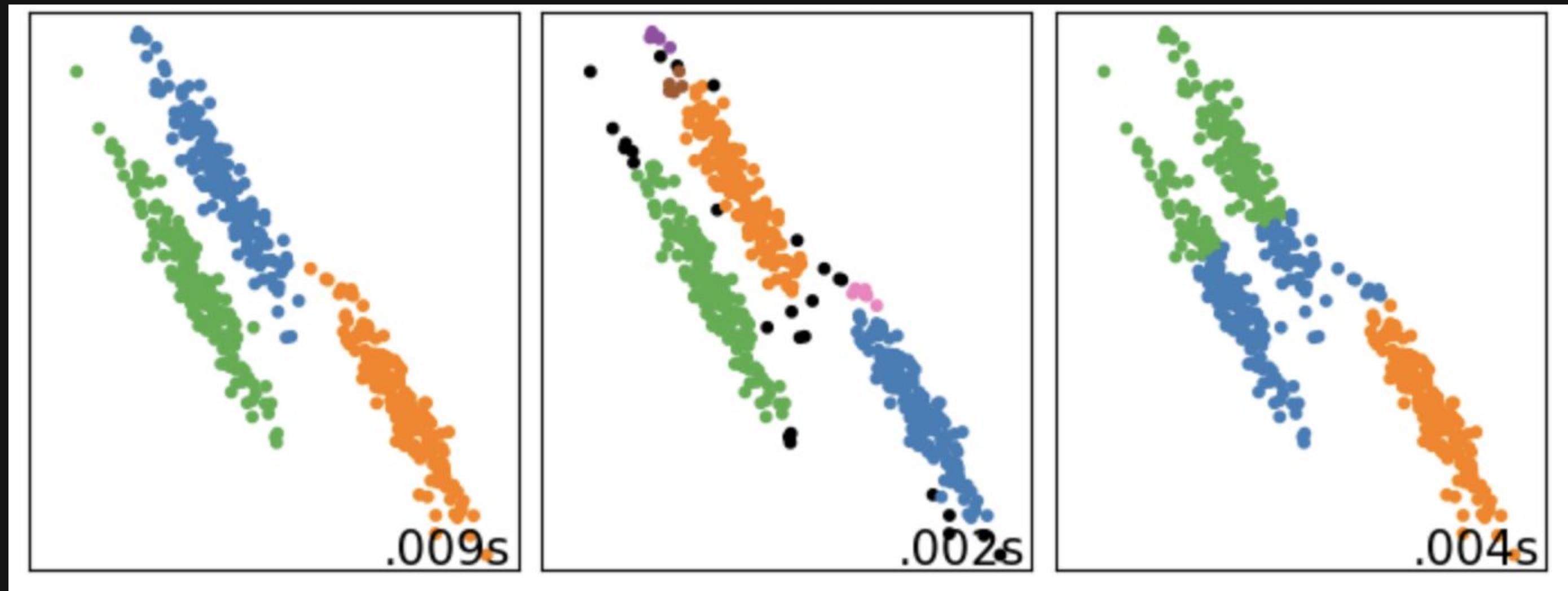


СОДЕРЖАНИЕ

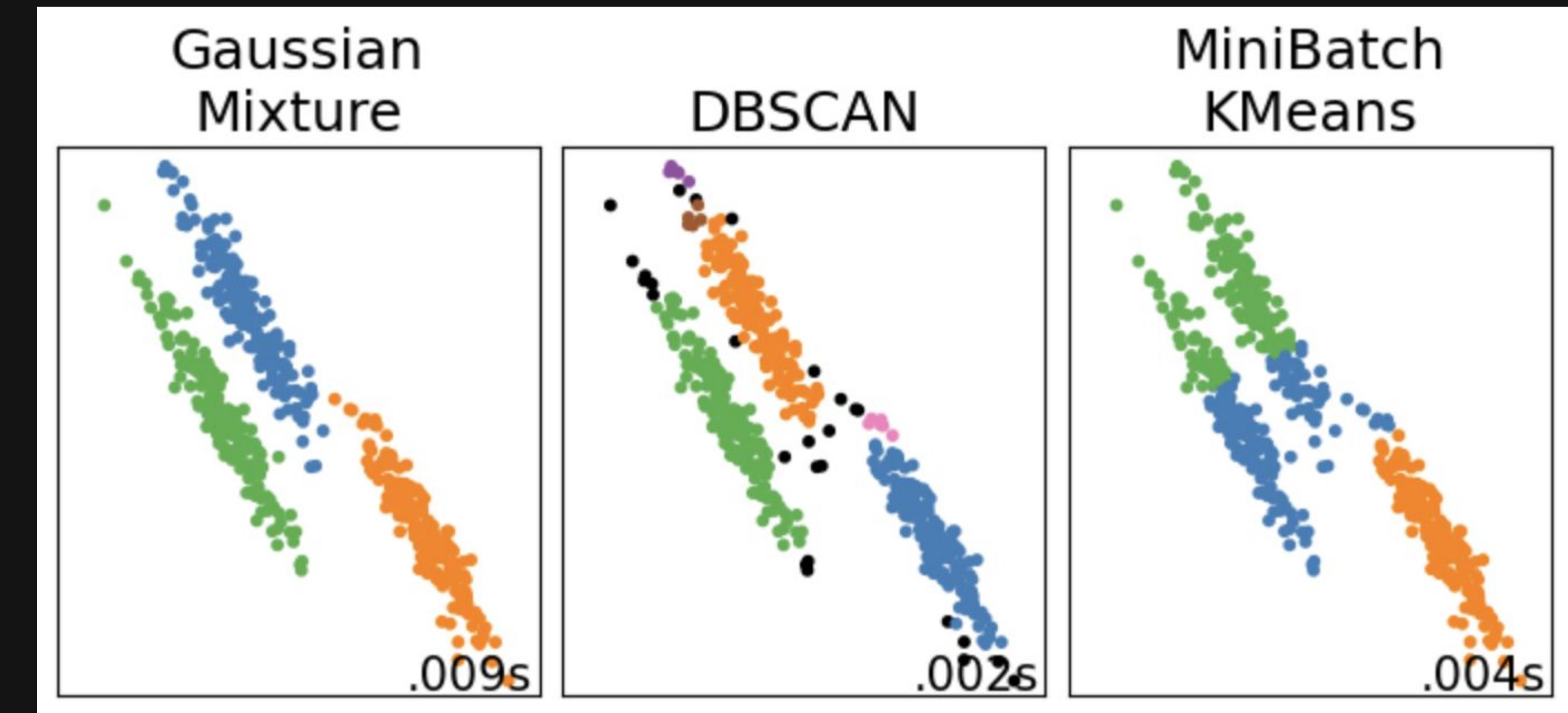
- Обучение без учителя:
кластеризация
- Алгоритмы кластеризации:
 - k-Means
 - EM algorithm
 - DBScan
- Оценка кластеризации
- Заключение



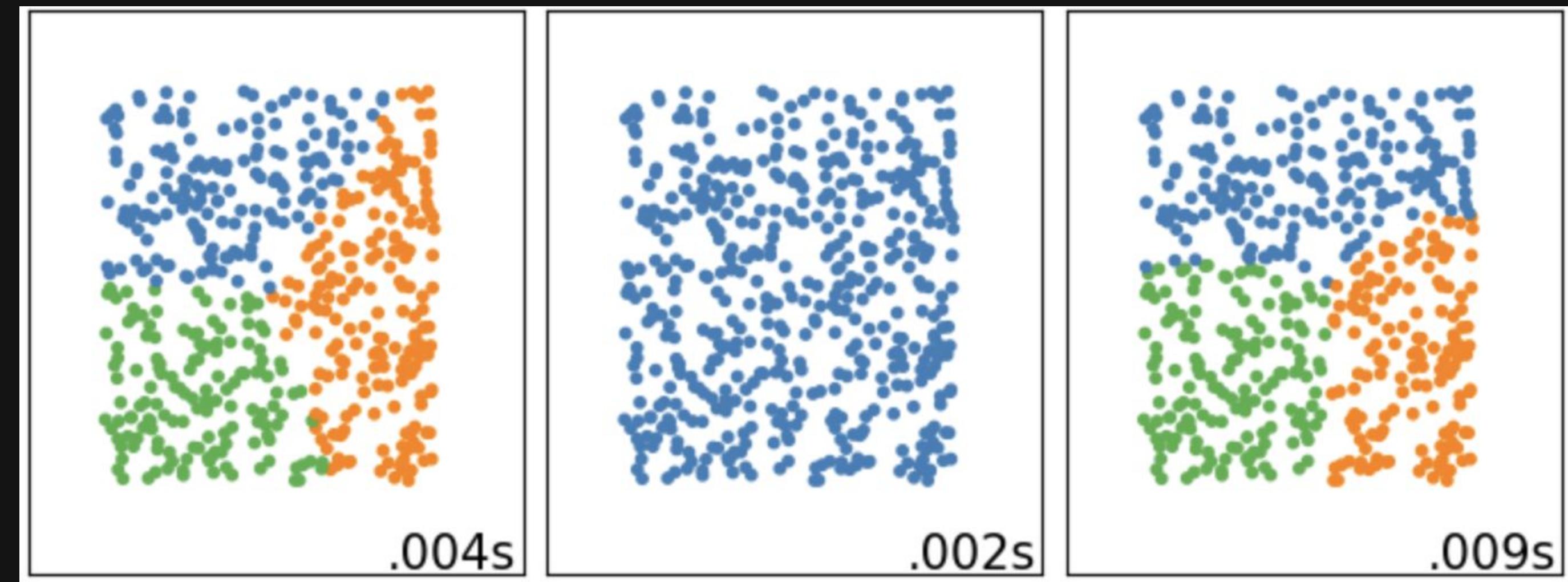
КВИЗ



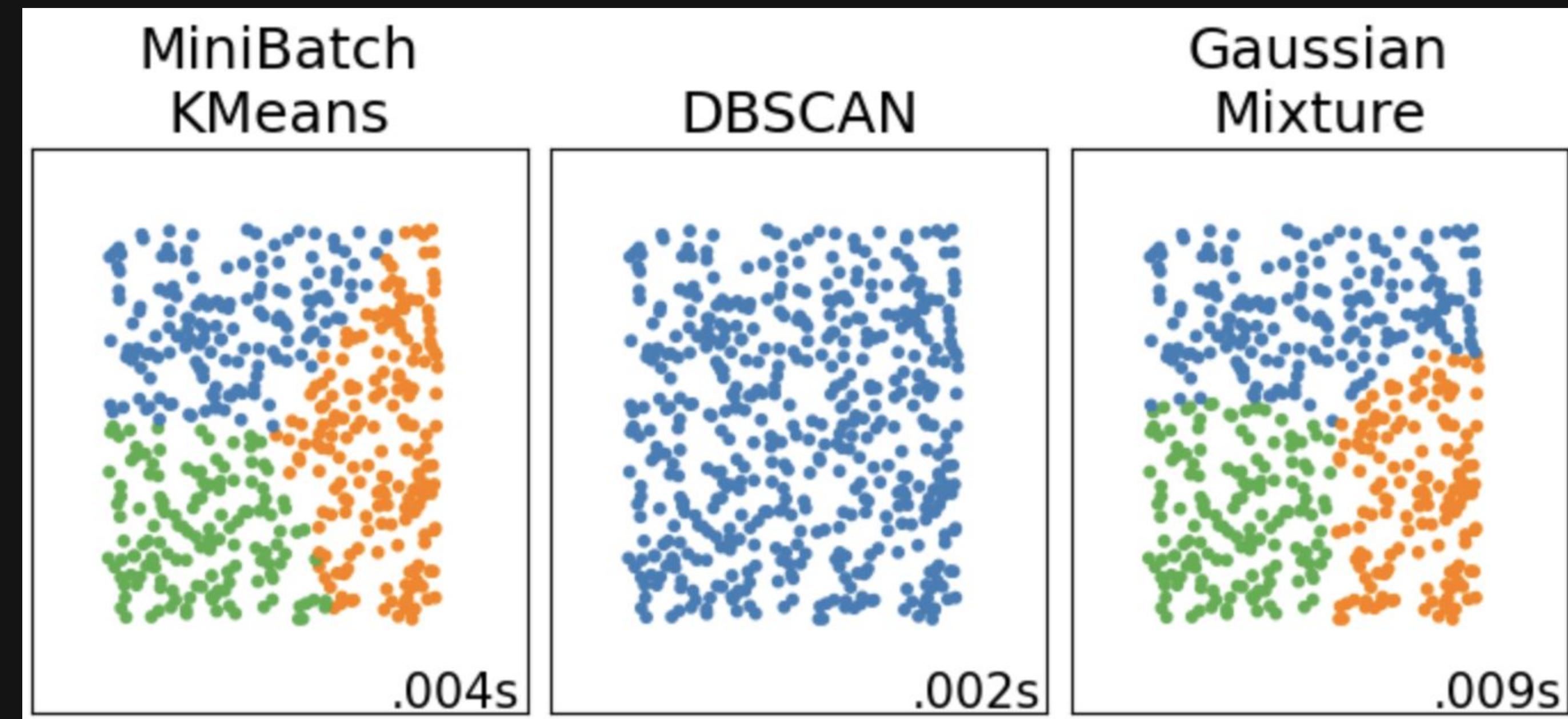
КВИЗ



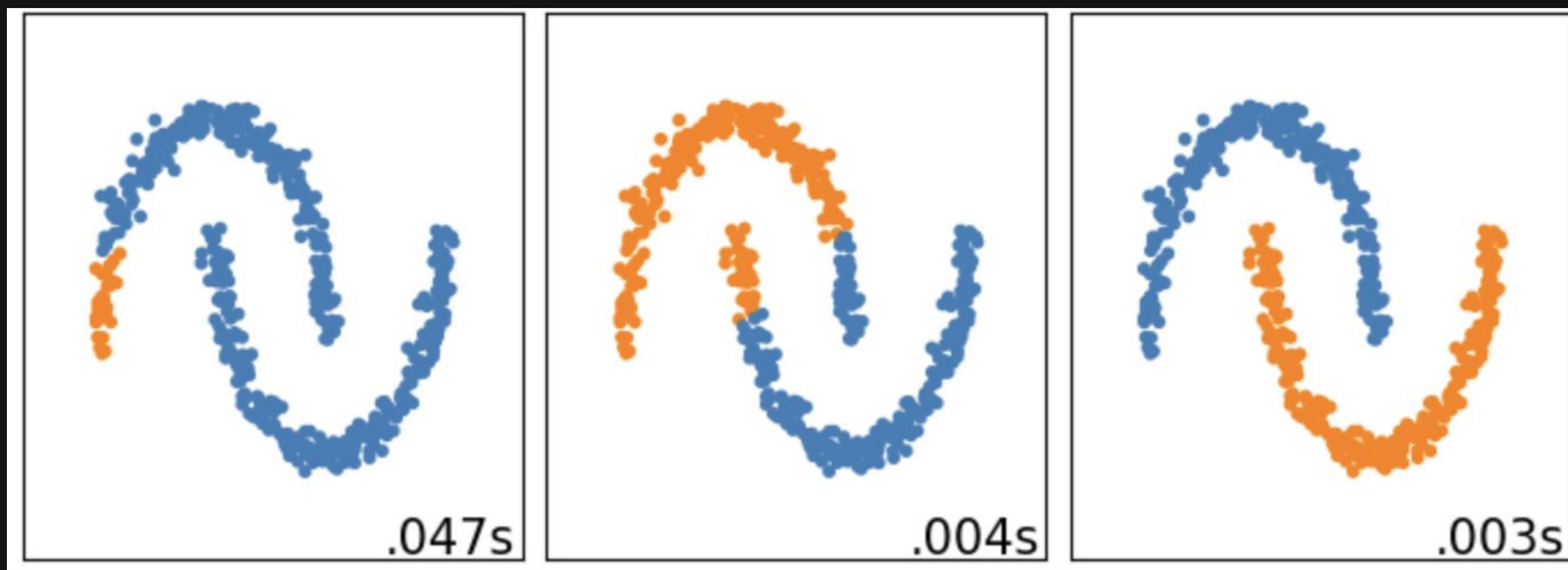
КВИЗ



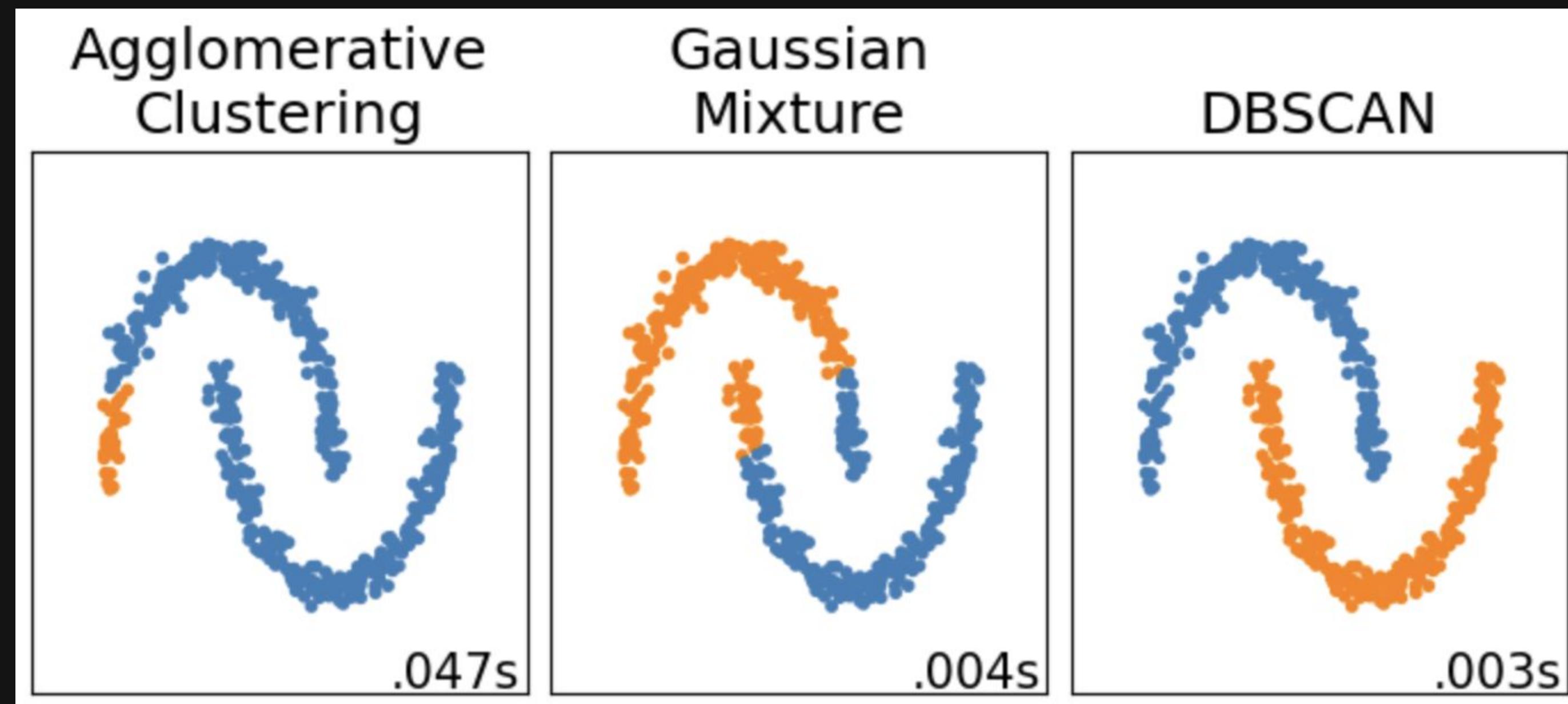
КВИЗ



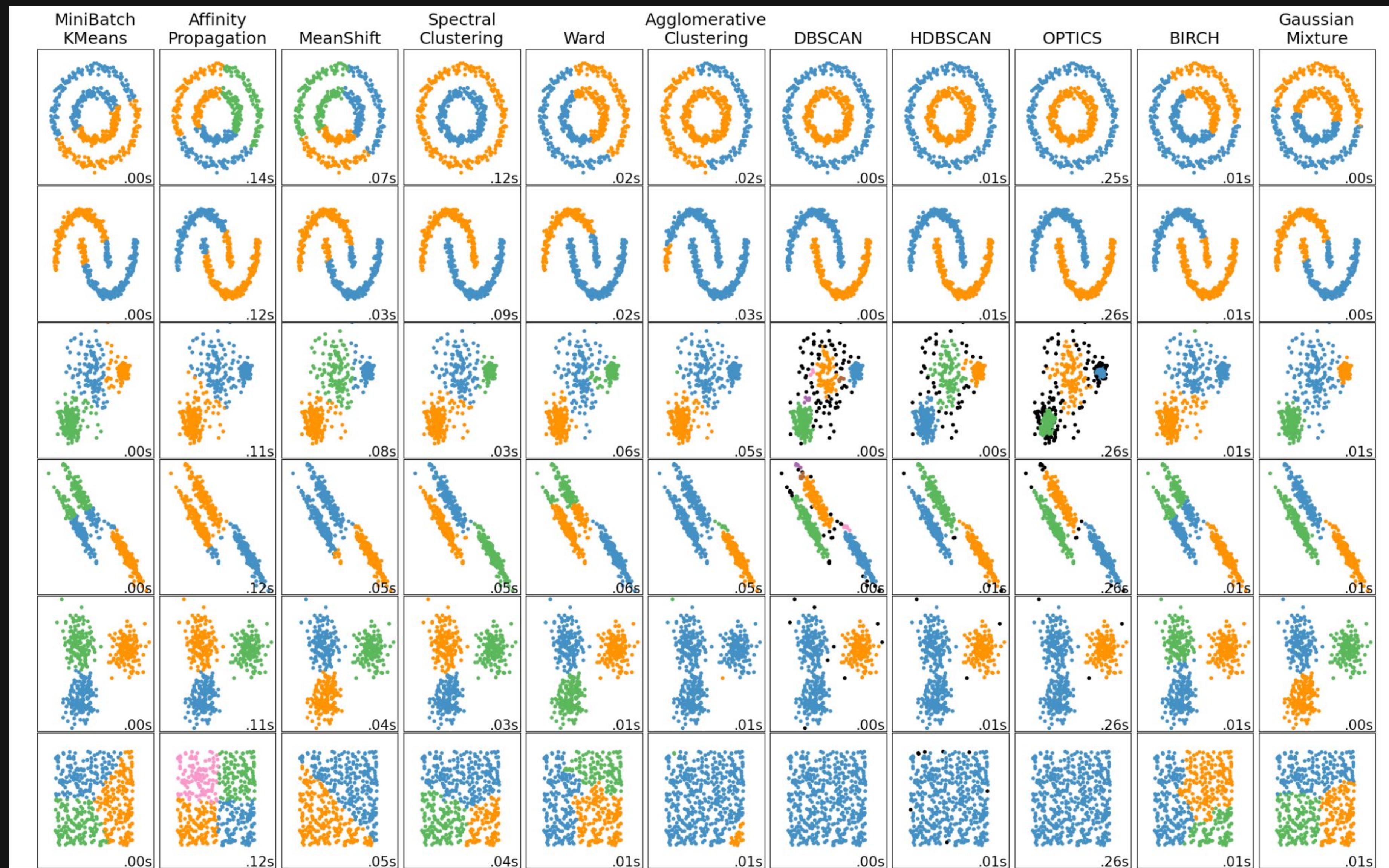
КВИЗ



КВИЗ



КВИЗ



ИТОГИ

k-Means

- Простой и быстрый
- Выпуклые примерно одинаковые кластера

EM

- Выдает вероятности принадлежности к кластеру
- Кластеры вытянутой формы с разной плотностью

DBSCAN

- Сам определяет количество кластеров
- Кластера различной формы

СОДЕРЖАНИЕ

- Обучение без учителя:
кластеризация
- Алгоритмы кластеризации:
 - k-Means
 - EM algorithm
 - DBScan
- Оценка кластеризации

ПЕРЕРЫВ

СОДЕРЖАНИЕ

- Обучение без учителя: задача понижения размерности
- Алгоритмы понижения размерности:
 - PCA
 - t-SNE
- Заключение

СОДЕРЖАНИЕ

- Обучение без учителя: задача понижения размерности
- Алгоритмы понижения размерности:
 - PCA
 - t-SNE
- Заключение



ЗАДАЧА ПОНИЖЕНИЯ РАЗМЕРНОСТИ

Кейс:

- Векторный поиск дублей картинок в недвижимости
- Модель, обученная на ImageNet (1000 классов)
- Понижение размерности эмбеддингов $4096 \rightarrow 128$ на основе наших данных

Какие плюсы и минусы?



ЗАДАЧА ПОНИЖЕНИЯ РАЗМЕРНОСТИ

Кейс:

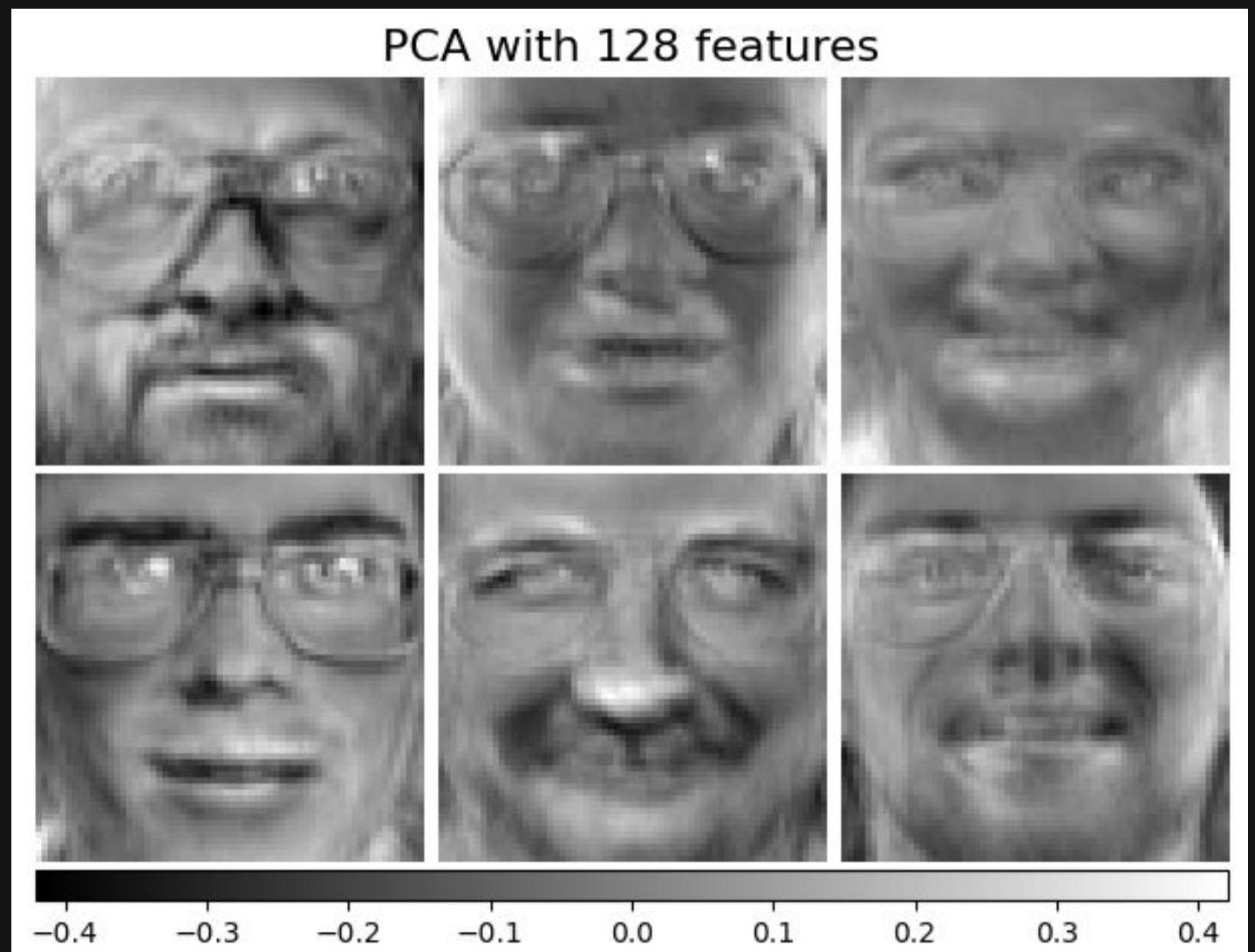
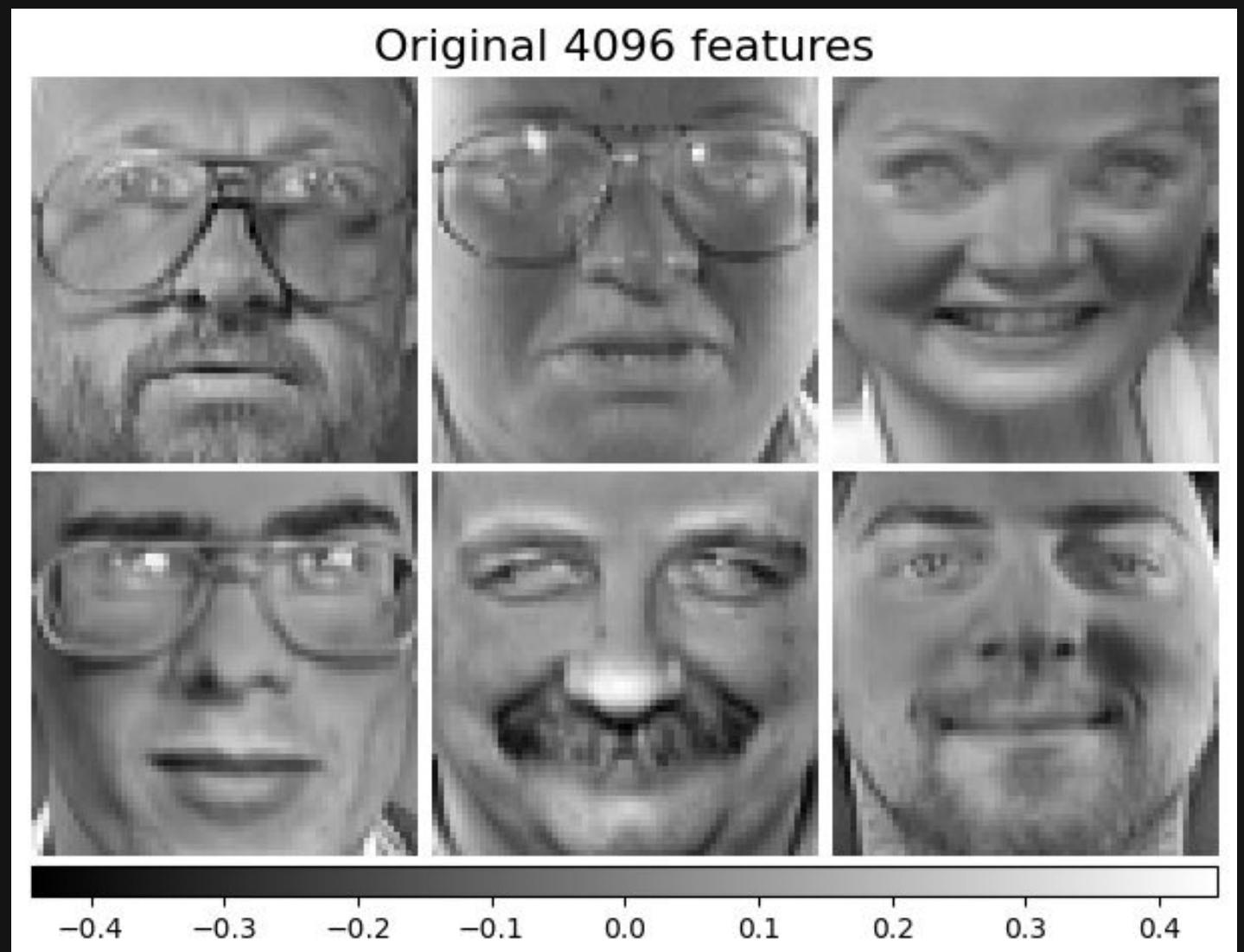
- Понижение размерности эмбеддингов 4096 -> 128

Плюсы:

- Быстрее поиск
- Меньше храним в памяти
- Избавляемся от избыточных признаков
- Снижаем проблему проклятия размерности

Минусы:

- Низже точность (возможно)

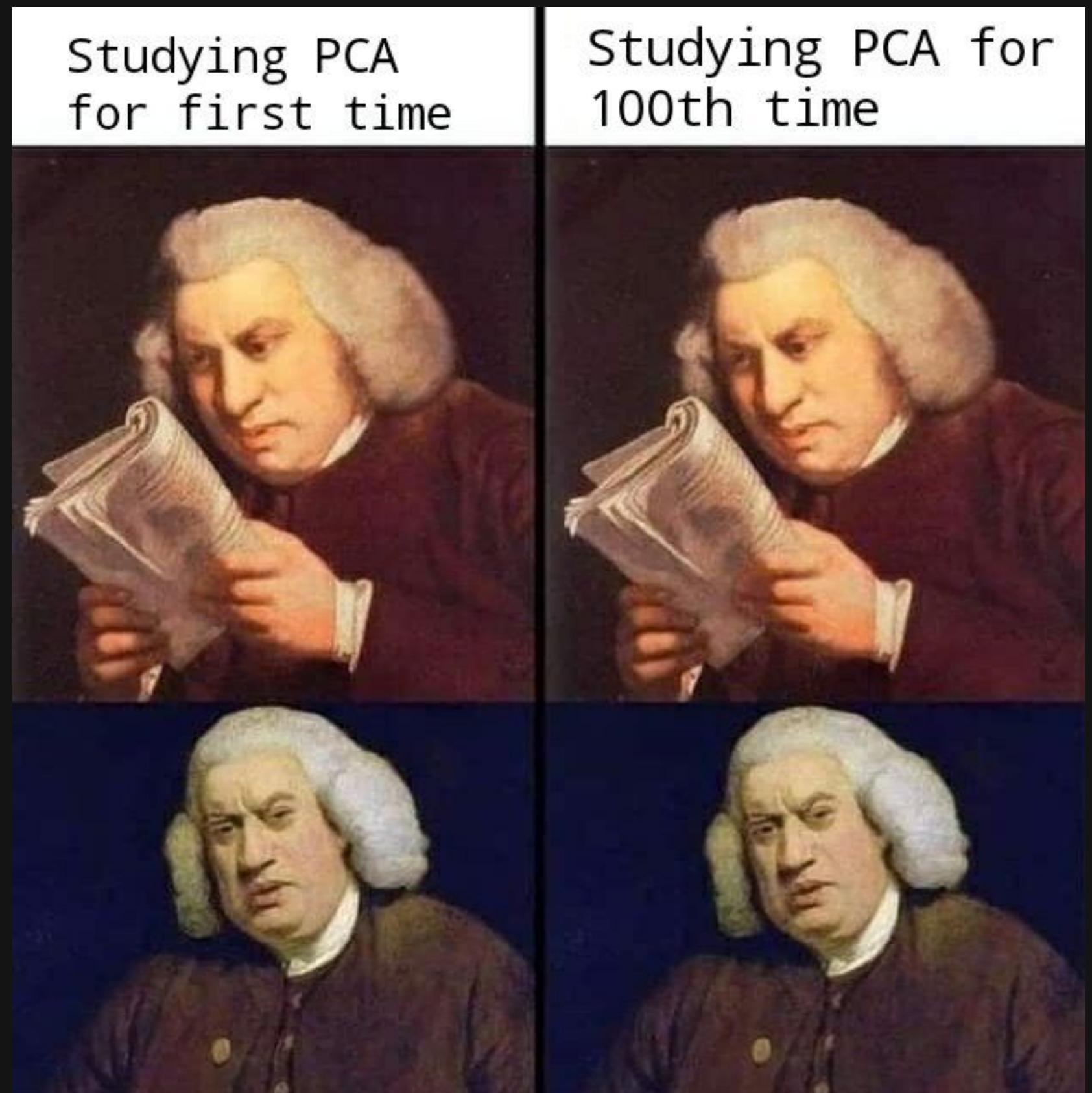


ЗАДАЧА ПОНИЖЕНИЯ РАЗМЕРНОСТИ

- Уменьшить размер датасета, потеряв при этом как можно меньше информации
- Можно получать модели меньшего размера, которые работают быстрее
- Заменять несколько зависимых признаков одним
- Визуализировать данные из высокоразмерных пространств

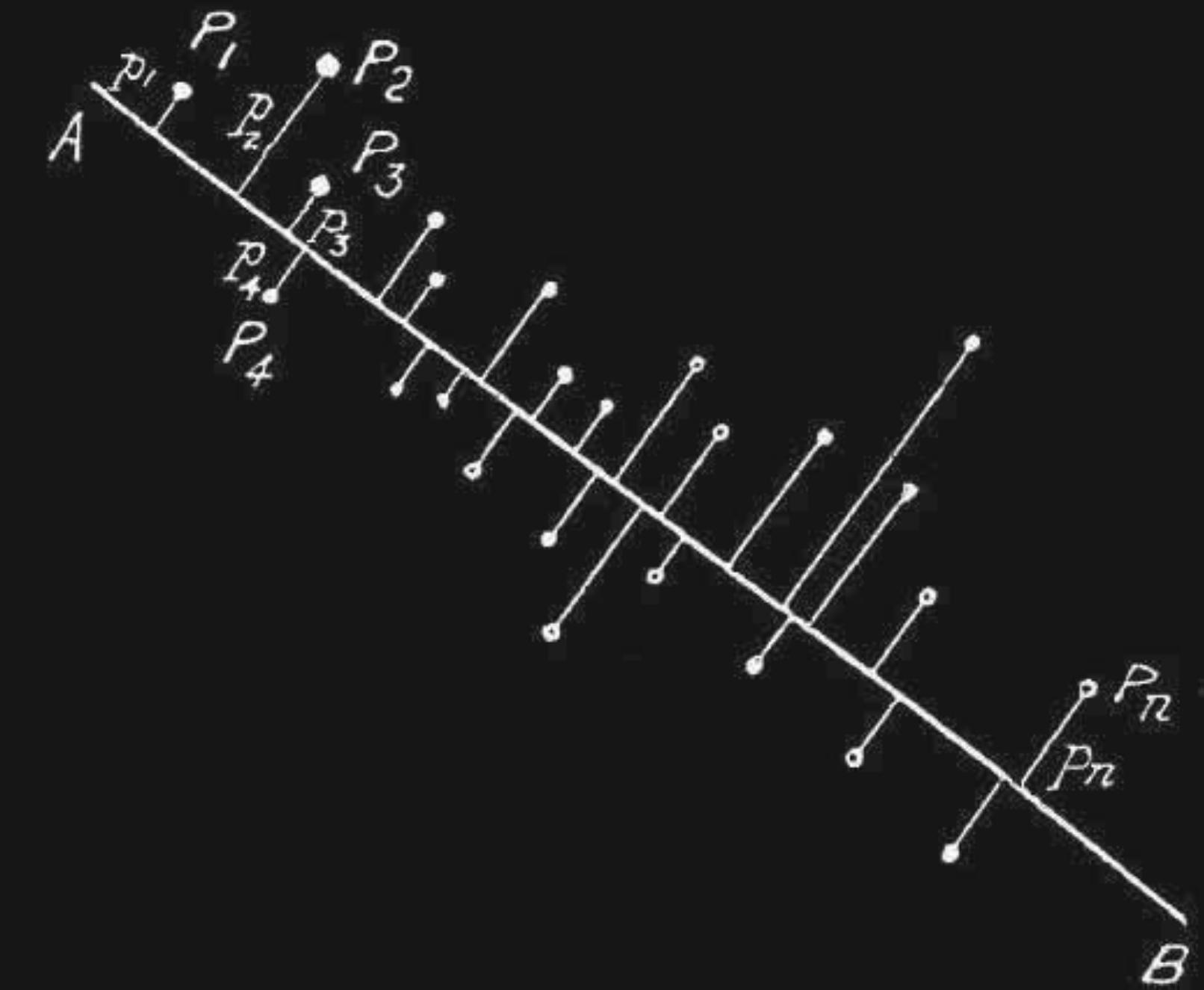
СОДЕРЖАНИЕ

- Обучение без учителя: задача понижения размерности
- Алгоритмы понижения размерности:
 - PCA
 - t-SNE
- Заключение



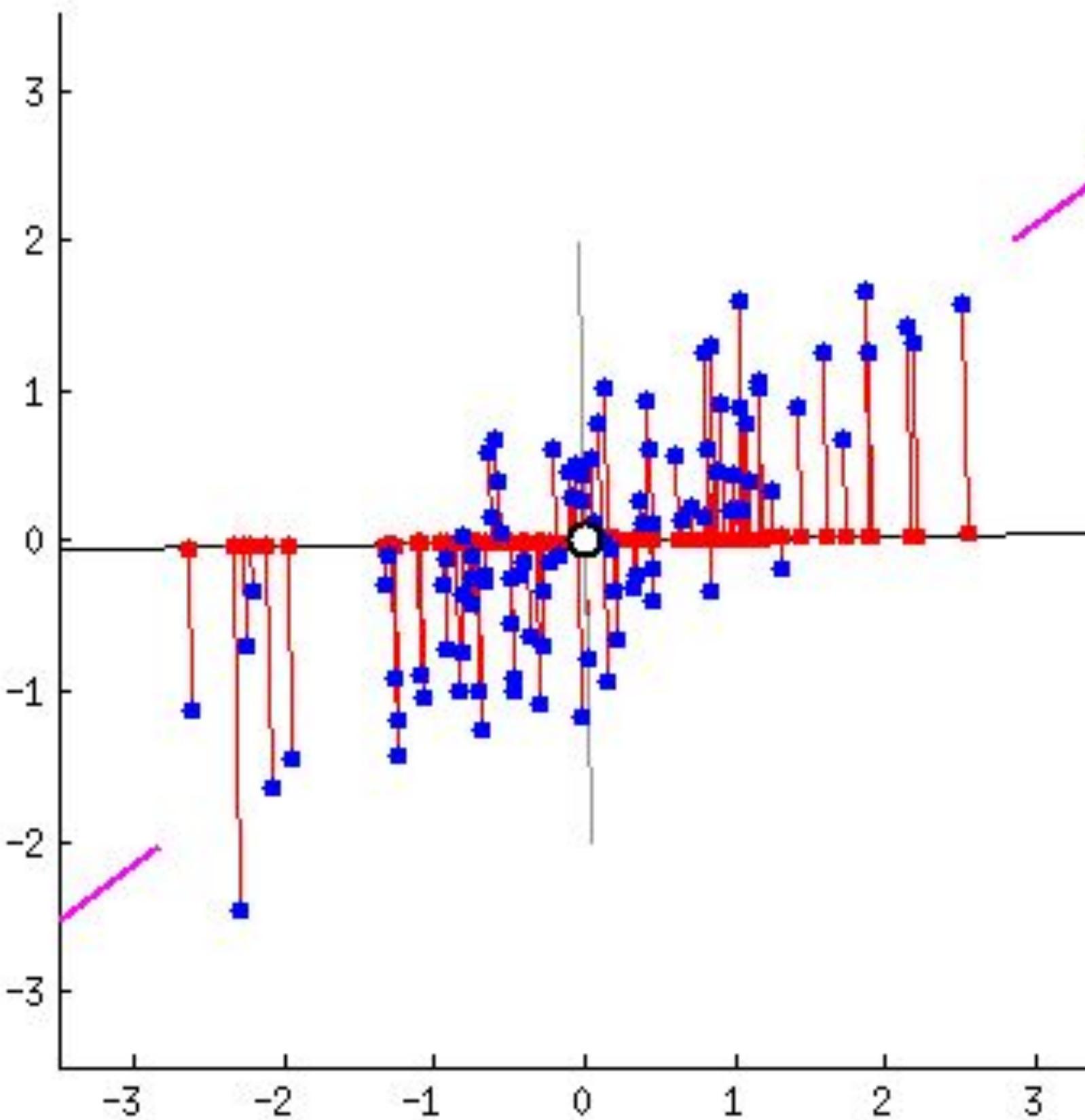
МЕТОД ГЛАВНЫХ КОМПОНЕНТ

- один из основных способов уменьшить размерность данных, потеряв наименьшее количество информации
- сводится к вычислению собственных векторов и собственных значений ковариационной матрицы исходных данных или к сингулярному разложению матрицы данных



МЕТОД ГЛАВНЫХ КОМПОНЕНТ

Principal component analysis



МЕТОД ГЛАВНЫХ КОМПОНЕНТ

Principal component analysis

Дисперсия — мера разнообразия признака

Ковариация — мера взаимного линейного разнообразия
признаков

Пусть \mathbf{C} — матрица ковариации

Найдем \mathbf{W} — матрицу собственных векторов матрицы ковариации

Тогда переход к пространству меньшей размерности: $\mathbf{X}_k = \mathbf{X}\mathbf{W}_k$

МЕТОД ГЛАВНЫХ КОМПОНЕНТ

Principal component analysis

$$var(X) = \frac{\sum_{i=1}^N (X - \bar{X})^2}{n - 1}$$

$$cov(X, Y) = \frac{\sum_{i=1}^N (X - \bar{X}) \cdot (Y - \bar{Y})}{n - 1} = \frac{\sum_{i=1}^N X \cdot Y}{n - 1}$$

Для РСА предполагаем, что данные отцентрированы, тогда

$$C = \begin{pmatrix} cov(X_1, X_1) & cov(X_1, X_2) \dots \\ cov(X_2, X_1) & cov(X_2, X_2) \dots \\ \vdots & \ddots \end{pmatrix} = \frac{X^T X}{n - 1}$$

МЕТОД ГЛАВНЫХ КОМПОНЕНТ

Principal component analysis

Спектральное разложение матрицы на матрицу собственных векторов и диаг. матрицу собственных чисел

$$C = W\Lambda W^{-1}$$

C — матрица ковариации

W — матрица собственных векторов

Λ — диагональная матрица собственных чисел

Проекция X на первые K собственных векторов

$$X_k = XW_k$$

МЕТОД ГЛАВНЫХ КОМПОНЕНТ

Principal component analysis

Сингулярное разложение (SVD) можно свести к PCA

$$X = U\Sigma V^*$$

U — унитарная матрица левых сингулярных векторов

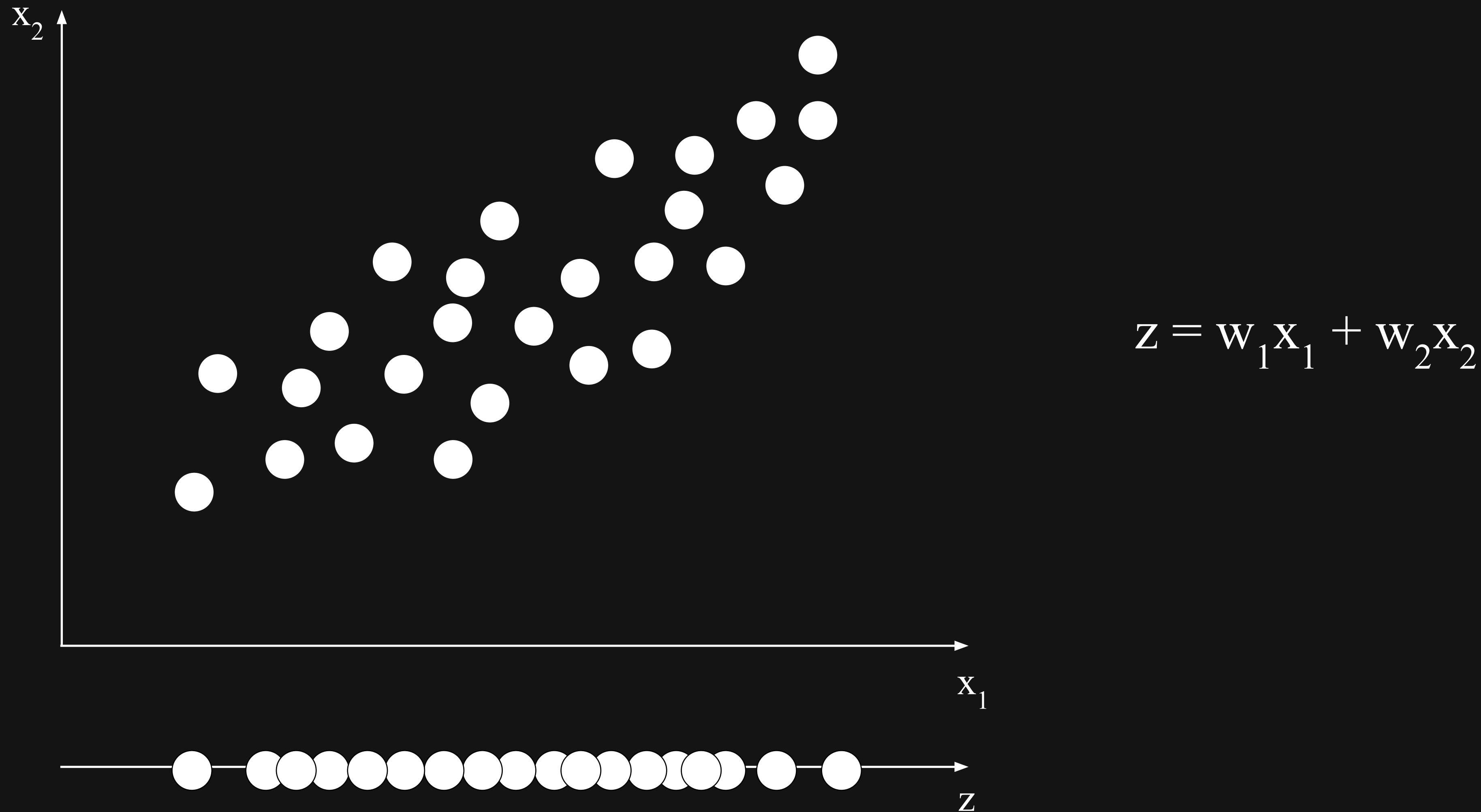
Σ — диагональная матрица сингулярных чисел

V^* — унитарная матрица правых сингулярных векторов

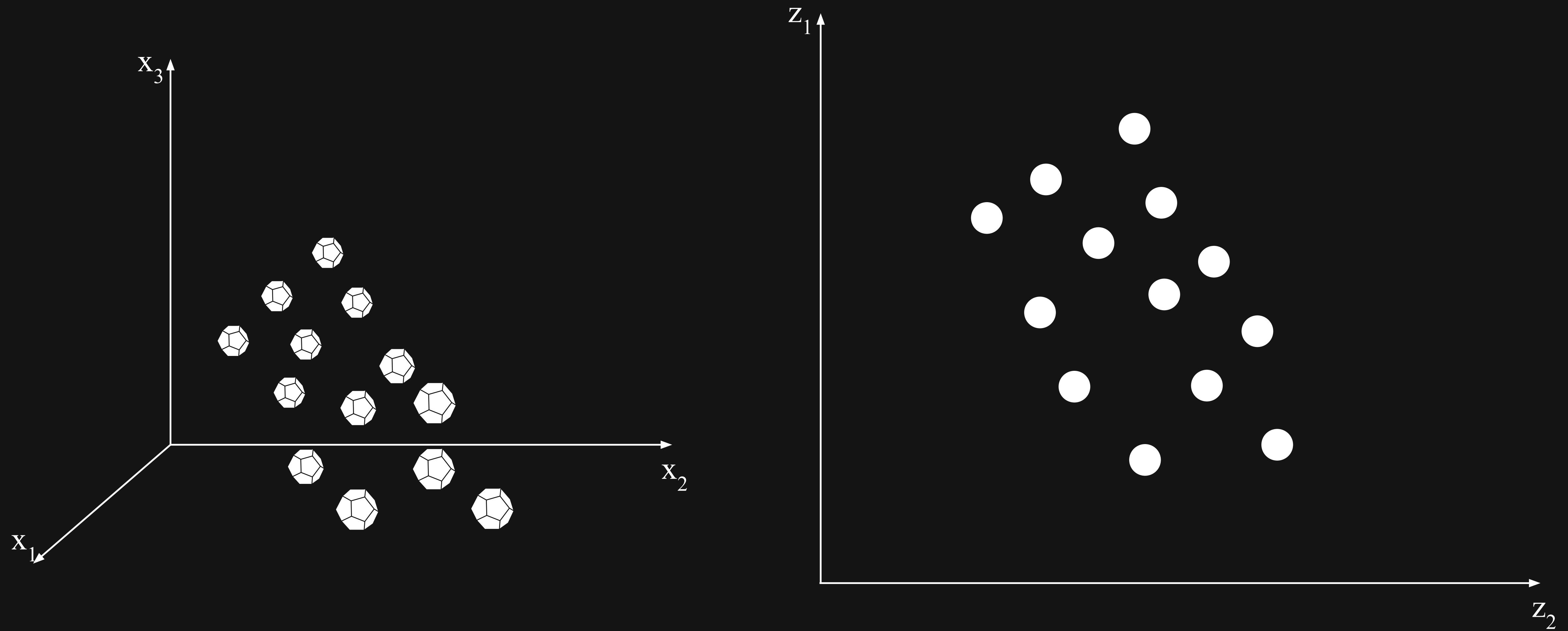
Преобразование для матрицы ковариации

$$\begin{aligned} C &= \frac{X^T X}{n-1} = \frac{V \Sigma U^T U \Sigma V^T}{n-1} = \frac{V \Sigma^2 V^T}{n-1} = \\ &= \frac{V \Sigma^2 V^{-1}}{n-1} = V \frac{\Sigma^2}{n-1} V^{-1} \end{aligned}$$

PCA (PRINCIPAL COMPONENT ANALYSIS): 2D \rightarrow 1D



PCA (PRINCIPAL COMPONENT ANALYSIS): 3D \rightarrow 2D

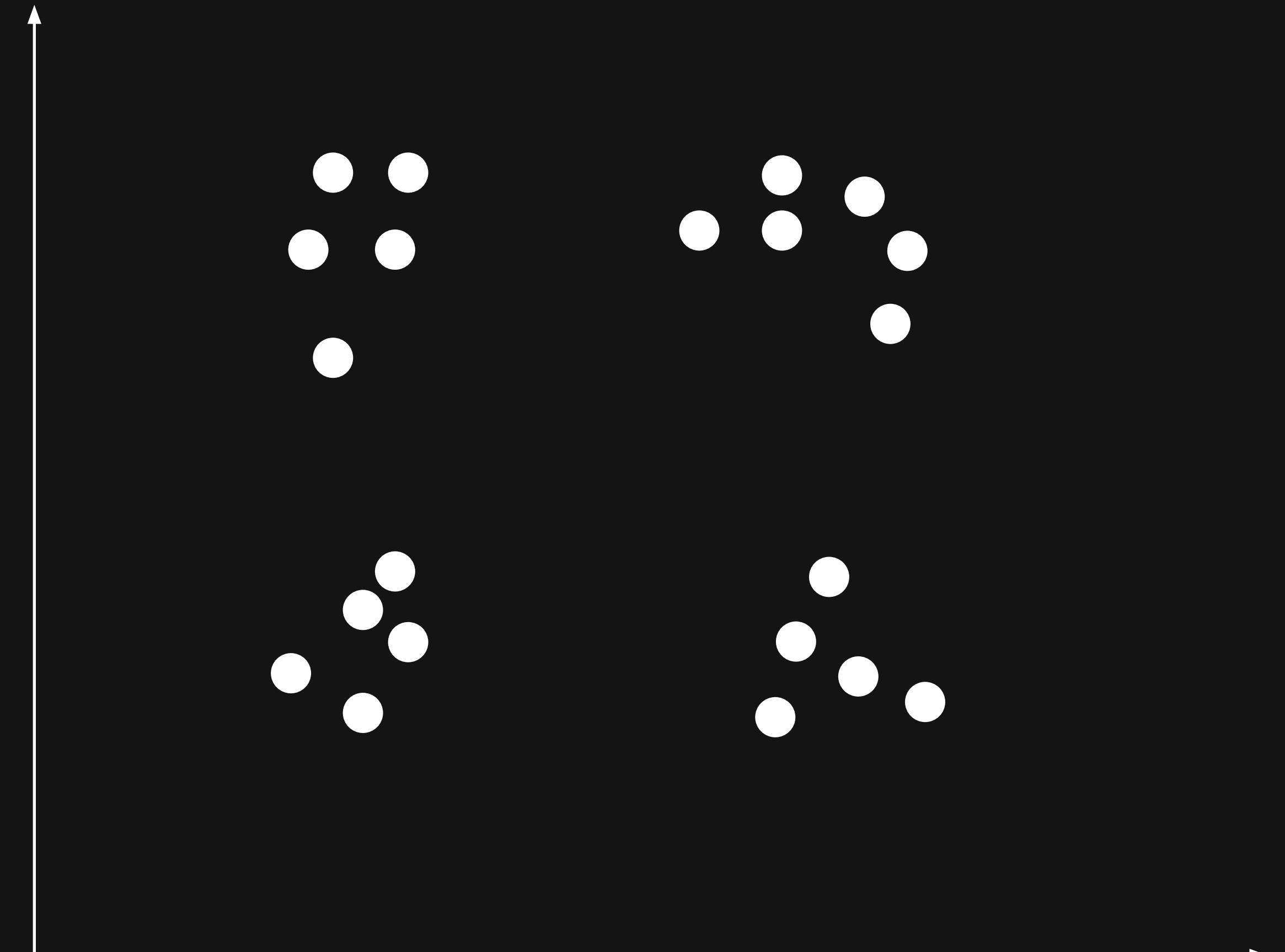


PCA ПРЕИМУЩЕСТВА

- Применим всегда (нет ограничений на распределение данных)
- Удаляет скореллированные признаки
- Уменьшает переобучение
- Улучшает производительность алгоритмов
- Нет гиперпараметров

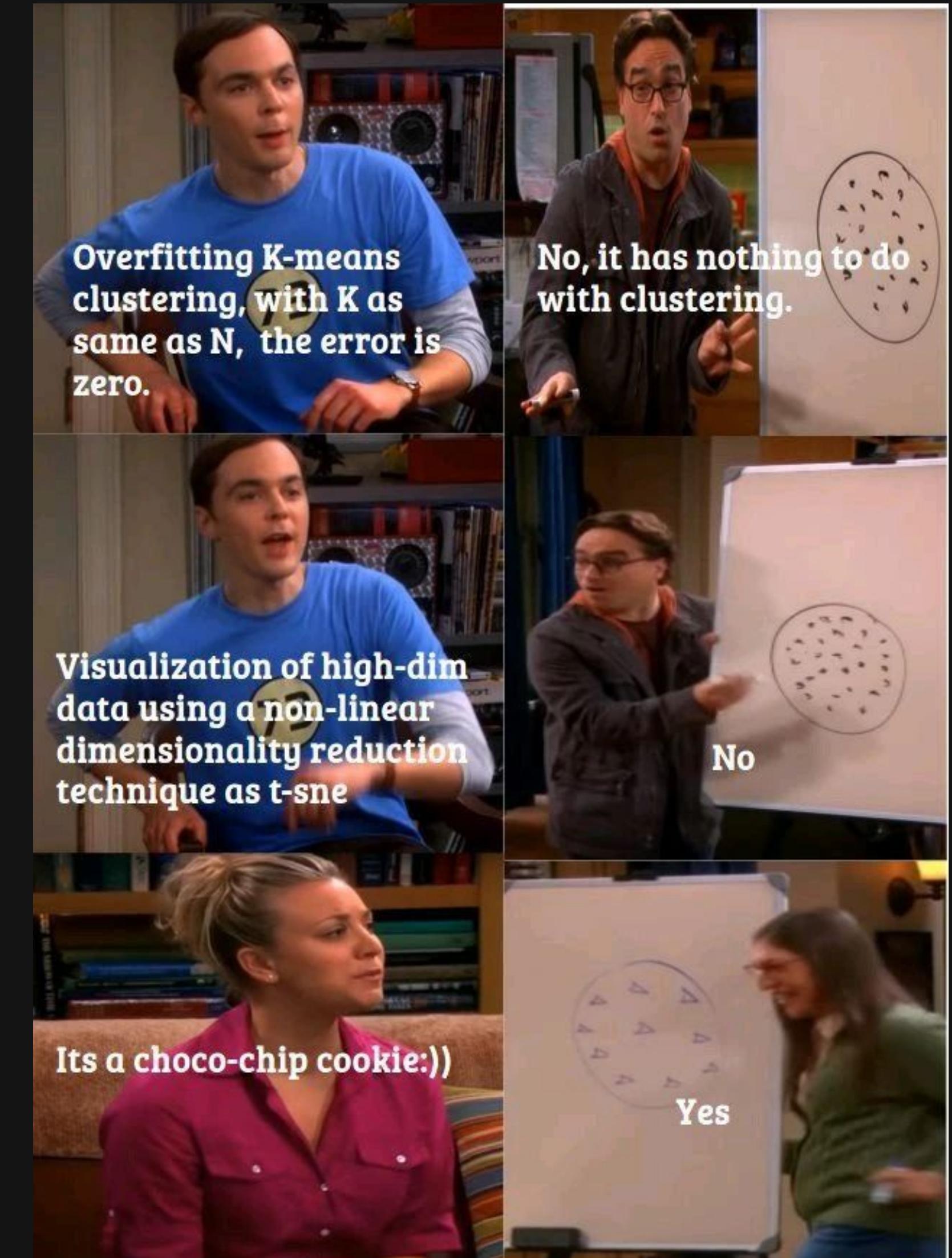
PCA НЕДОСТАТКИ

- Необходимость нормализации признаков
- Потеря информации
- Признаки менее интерпретируемые
- Хорошо работает только для линейных зависимостей (решается kernel-PCA)
- Плохо работает если есть какой-то микс сигналов, сработает плохо (решается ICA)
- Плохо работает с разреженными данными (решается Truncated SVD / LSA)

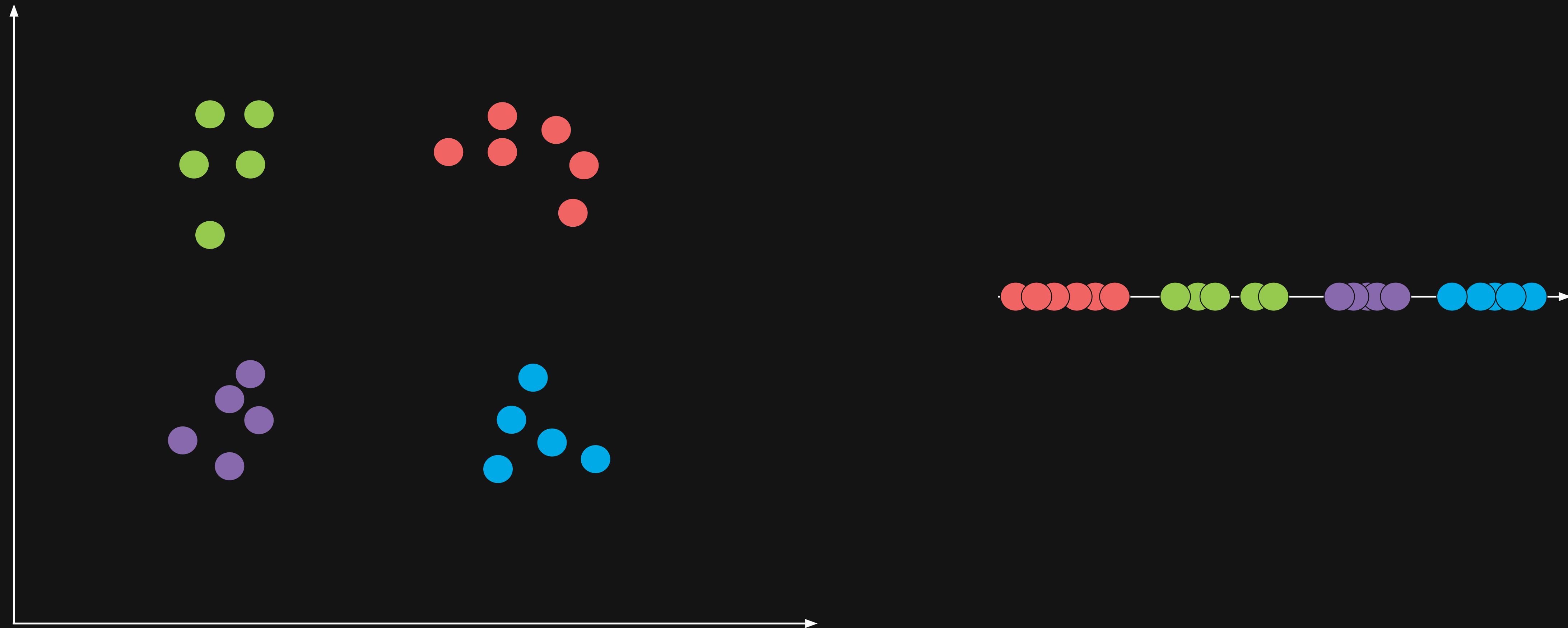


СОДЕРЖАНИЕ

- Обучение без учителя: задача понижения размерности
- Алгоритмы понижения размерности:
 - PCA
 - t-SNE
- Заключение



СТОХАСТИЧЕСКОЕ ВЛОЖЕНИЕ СОСЕДЕЙ С Т-РАСПРЕДЕЛЕНИЕМ (T-SNE)



СТОХАСТИЧЕСКОЕ ВЛОЖЕНИЕ СОСЕДЕЙ С Т-РАСПРЕДЕЛЕНИЕМ (T-SNE)

- Метод визуализации данных
- Конвертируем Евклидово расстояние в условные вероятности:
 - $p_{j|i}$ — вероятность, что точка x_i выберет в качестве своего соседа точку x_j среди остальных точек данных для пространства высокой размерности
 - $q_{i|j}$ для пространства низкой размерности

$$p_{j|i} = \frac{\exp \frac{|x_i - x_j|^2}{2\sigma_i^2}}{\sum_{k \neq i} \exp \frac{|x_i - x_k|^2}{2\sigma_i^2}}$$
$$q_{ij} = \frac{(1 + |y_i - y_j|^2)^{-1}}{\sum_{l \neq i} (1 + |y_i - y_l|^2)^{-1}}$$

$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2n}$$

СТОХАСТИЧЕСКОЕ ВЛОЖЕНИЕ СОСЕДЕЙ С Т-РАСПРЕДЕЛЕНИЕМ (T-SNE)

- Вероятность для точки найти соседа падает с увеличением расстояния от точки в соответствии
 - с нормальным распределением для высокой размерности (x_i)
 - с распределением Стьюдента для низкой размерности (y_i)
- Уменьшаем разницу в распределении вероятностей в высокой и низкой размерности с помощью дивергенции Кульбака-Лейблера

АЛГОРИТМ T-SNE

- Вычислить попарные вероятности $p_{j|i}$ с перплексией P_{perp}
- $p_{ij} = (p_{j|i} + p_{i|j}) / 2n$ - из условных в совместные вероятности
- Инициализация Y случайными значениями
- До схождения:
 - Вычислить q_{ij}
 - Вычислить градиент
 - Сделать градиентный шаг

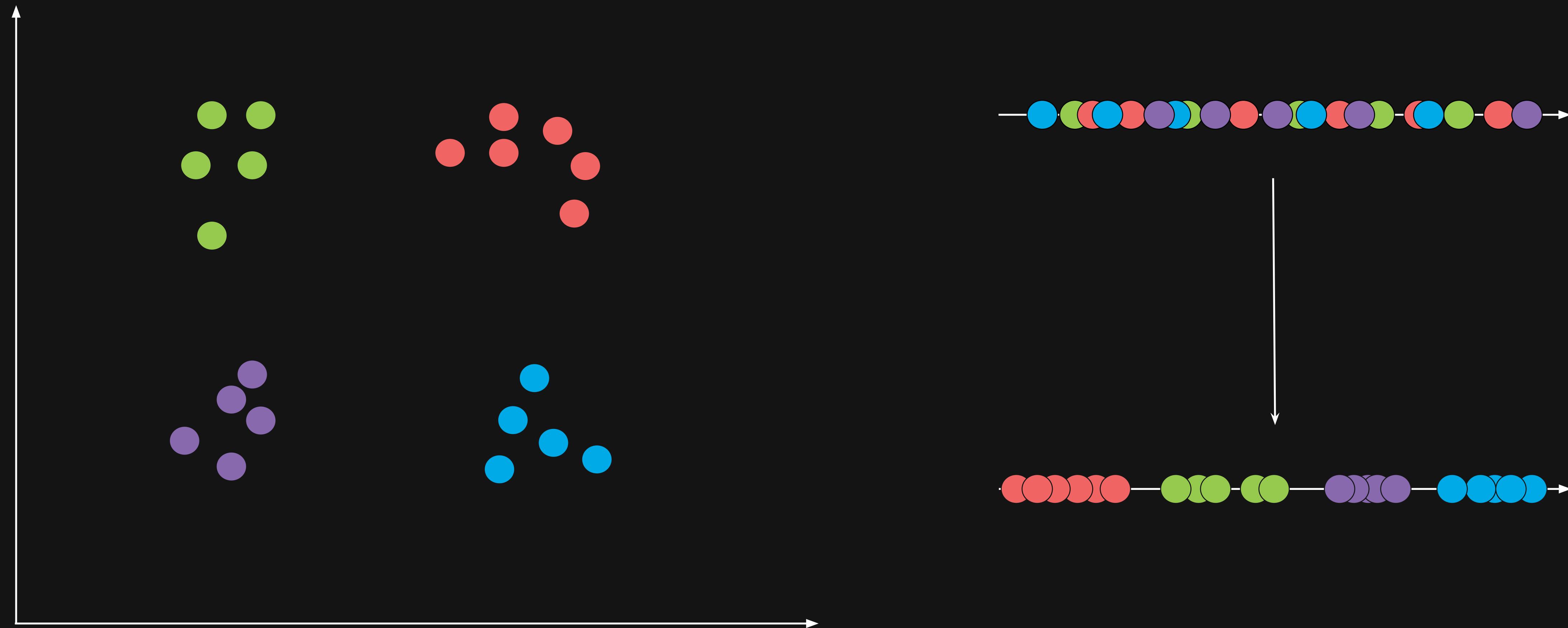
$$p_{ij} = \frac{p_{j|i} + p_{i|j}}{2n}$$

$$KL(P||Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}$$

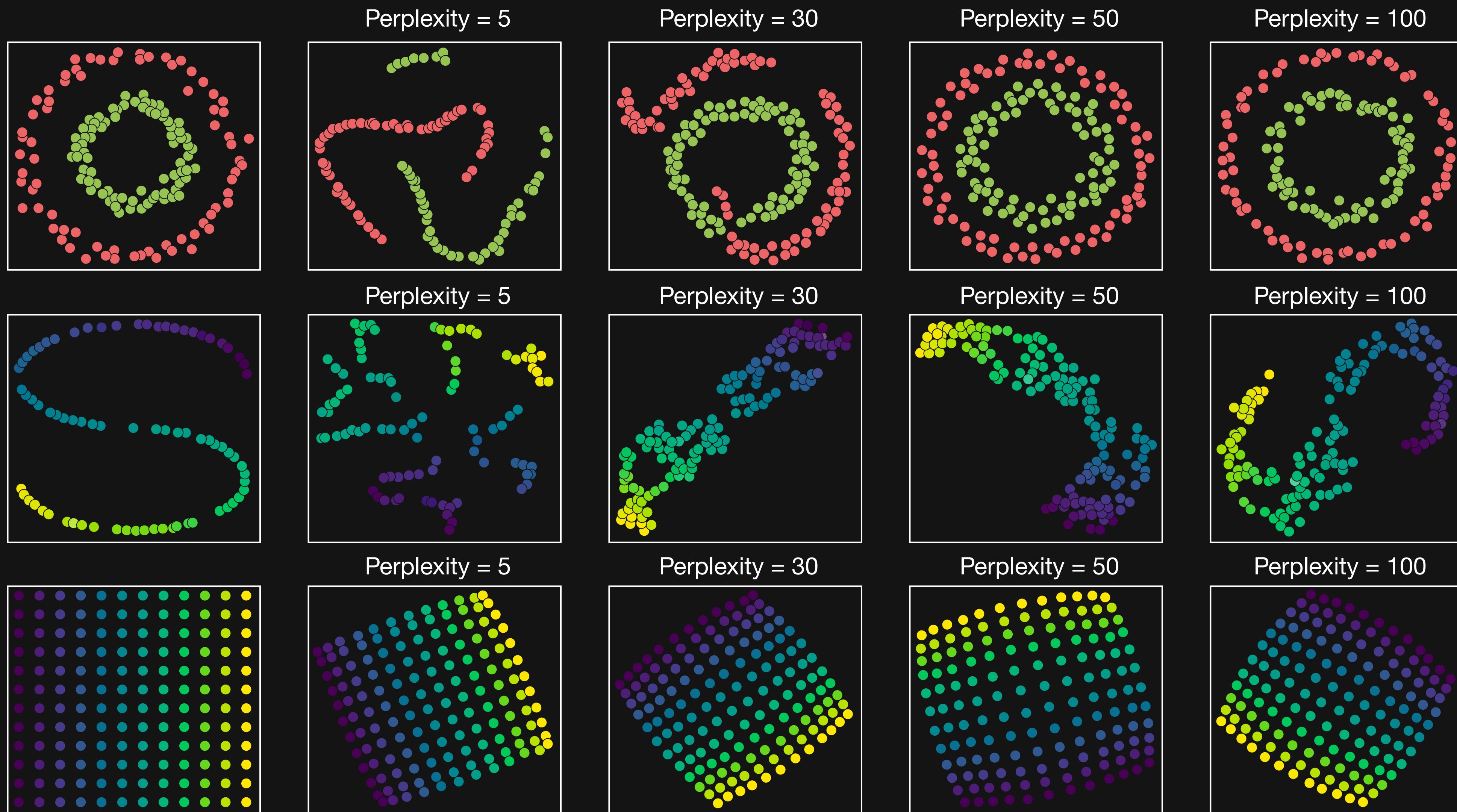
$$\frac{\delta KL}{\delta y_i} = 4 \sum_j (p_{ij} - q_{ij})(y_i - y_j)(1 + \|y_i - y_j\|^2)^{-1}$$

$$Y(t) = Y(t-1) + \eta \delta \frac{KL}{\delta y} + \alpha(t)(Y(t-1) - Y(t-2))$$

СТОХАСТИЧЕСКОЕ ВЛОЖЕНИЕ СОСЕДЕЙ С Т-РАСПРЕДЕЛЕНИЕМ (T-SNE)

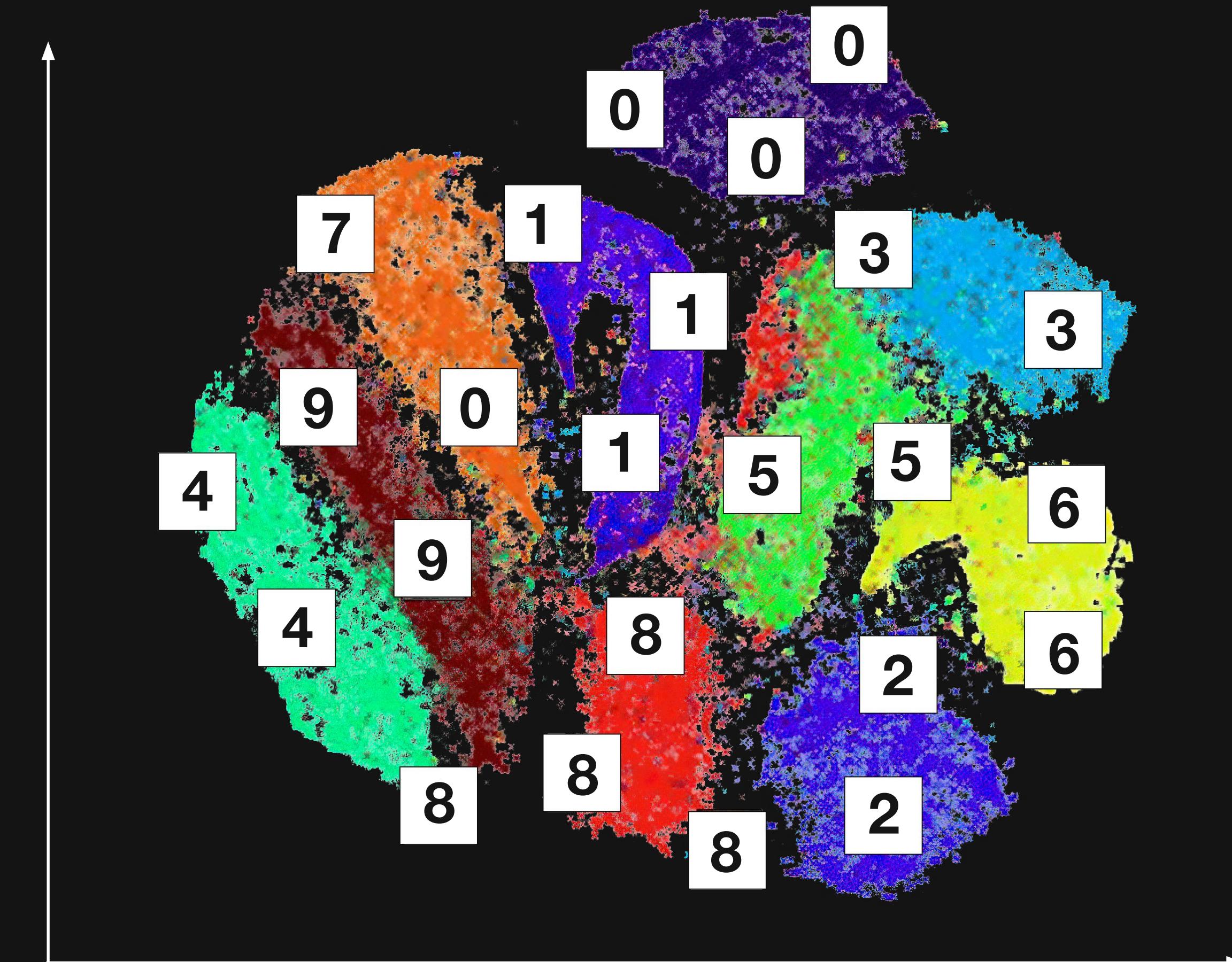


ПЕРПЛЕКСИЯ (PERPLEXITY)



ПОНИЖЕНИЕ РАЗМЕРНОСТИ: (T-DISTRIBUTED STOCHASTIC NEIGHBOR EMBEDDING)

```
array([
    [0., 0., 5., ..., 0., 0., 0.],
    [0., 0., 0., ..., 10., 0., 0.],
    [0., 0., 0., ..., 16., 9., 0.],
    ...,
    [0., 0., 1., ..., 6., 0., 0.],
    [0., 0., 2., ..., 12., 0., 0.],
    [0., 0., 10., ..., 12., 1., 0.]])
```



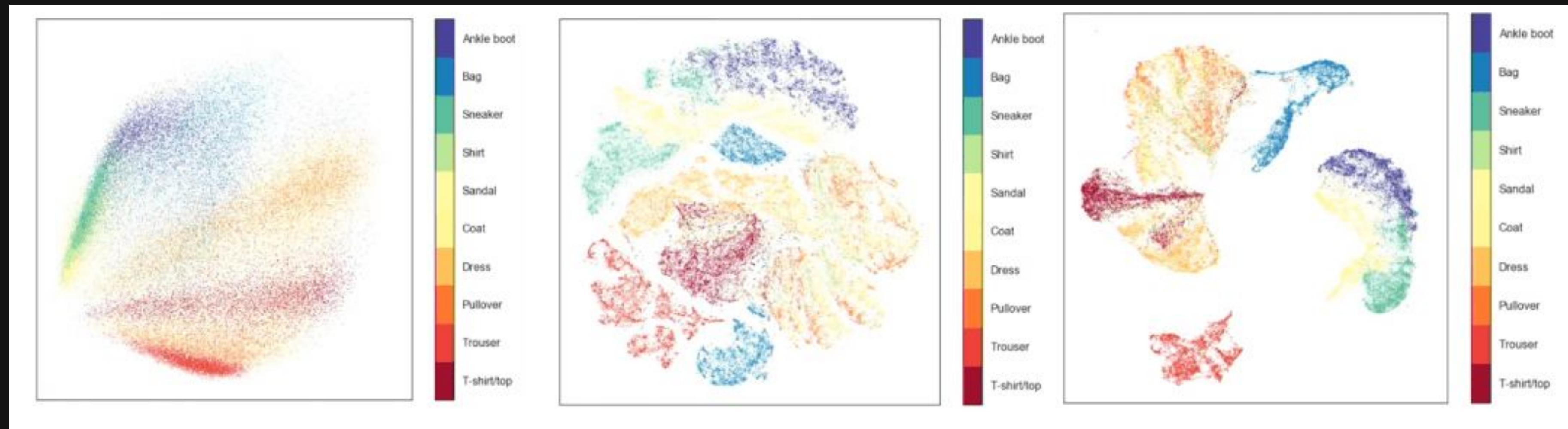
t-SNE НЕДОСТАТКИ

- долго работает
- сложная метрика минимизации
- нестабилен
- размеры полученных сгустков могут ничего не значить
- расстояния между кластерами могут ничего не значить
- нельзя восстановить исходный вид матрицы
- нельзя добавить новый объект без полного пересчета

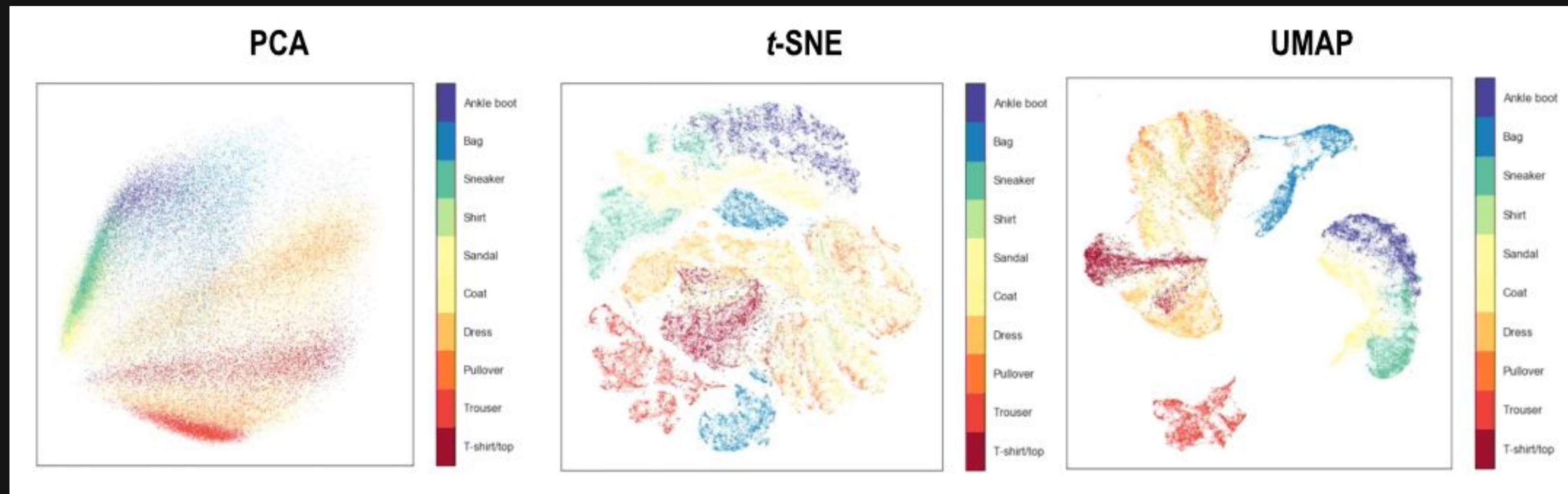
СОДЕРЖАНИЕ

- Обучение без учителя: задача понижения размерности
- Алгоритмы понижения размерности:
 - PCA
 - t-SNE
- Заключение

КВИЗ



КВИЗ



Performance Comparison		Performance Comparison	
	t-SNE	UMAP	UMAP speed up over t-SNE
COIL20	20 seconds	7 seconds	3x
MNIST	22 minutes	98 seconds	13x
Fashion MNIST	15 minutes	78 seconds	11x
GoogleNews	4.5 hours	14 minutes	19x

ИТОГИ

PCA

- Для снижения размерности с наименьшей потерей информации
- Относительно быстрый
- Нет гиперпараметров

t-SNE

- Для визуализации
- Применим для нелинейных зависимостей

ЗАКЛЮЧЕНИЕ



СОДЕРЖАНИЕ

- Обучение без учителя: задача понижения размерности
- Алгоритмы понижения размерности:
 - PCA
 - t-SNE
- Заключение

СПАСИБО ЗА ВНИМАНИЕ!