

ВВЕДЕНИЕ В КОМПЬЮТЕРНОЕ ЗРЕНИЕ

Лекция № 7

Сегментация изображений



План лекции

1. Введение в кластеризацию и сегментацию

2. Agglomerative clustering

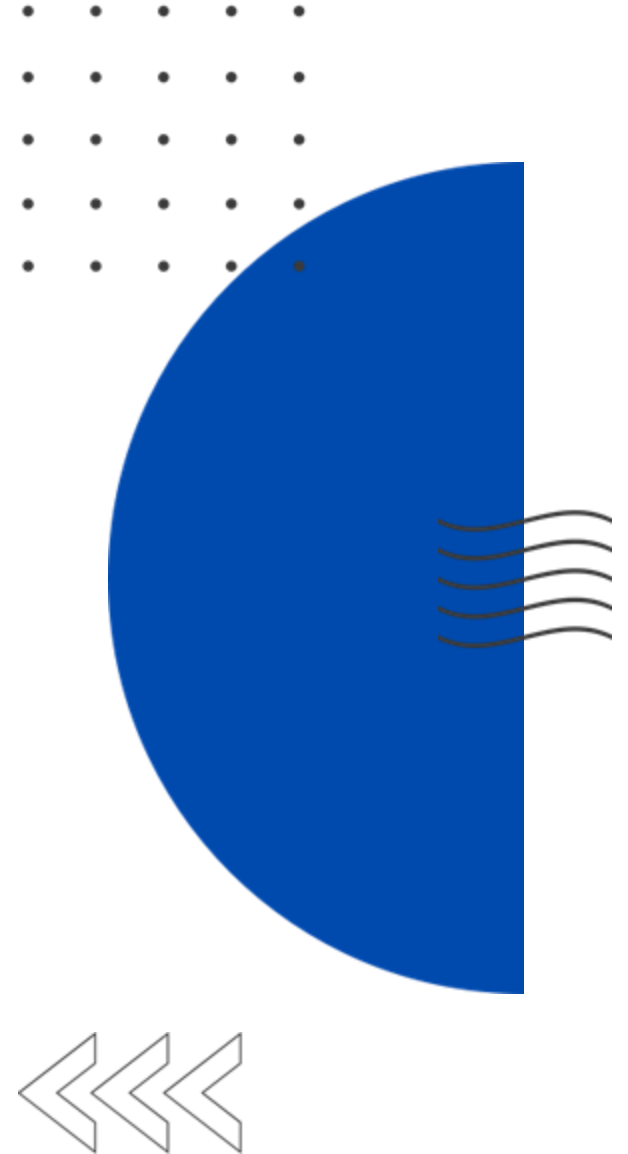
3. Mean-shift clustering

4. Метрики качества сегментации
Dice, Jaccard



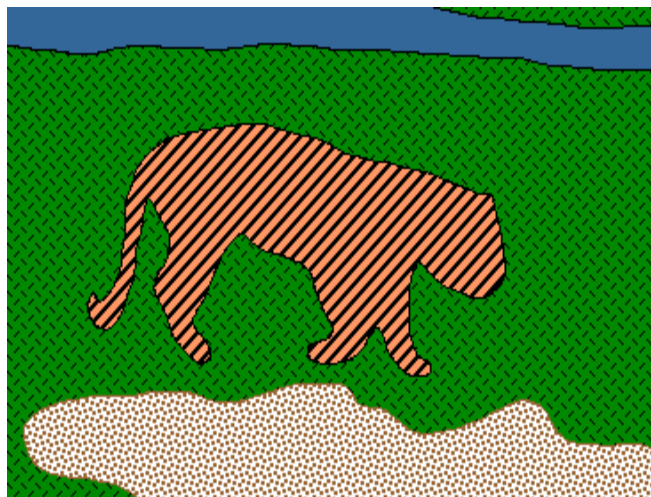
01

Введение в кластеризацию и сегментацию



Сегментация изображений

- Цель: определить похожие группы пикселей



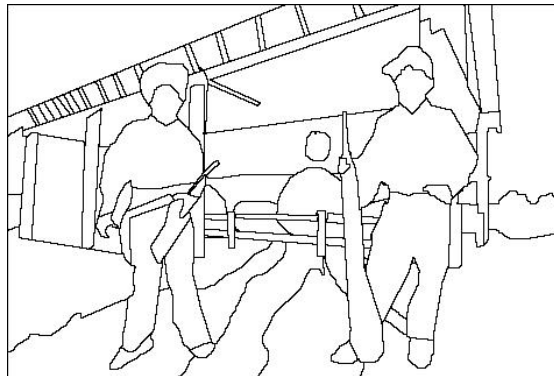
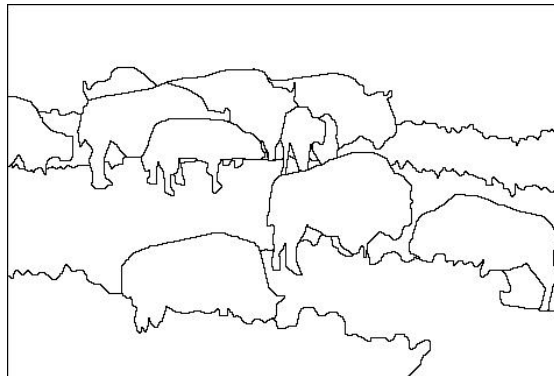
Цель сегментации

- Разделите изображение на схожие "объекты"

Image



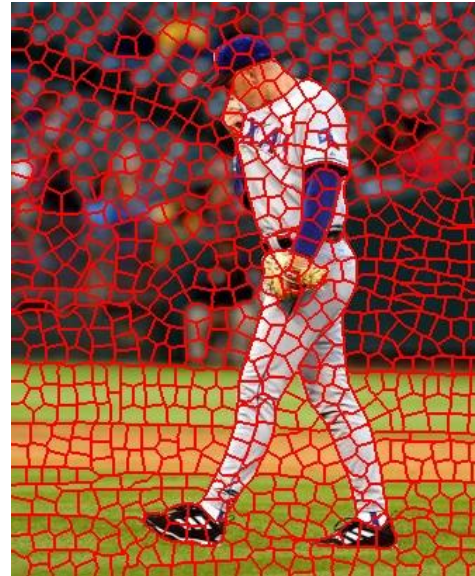
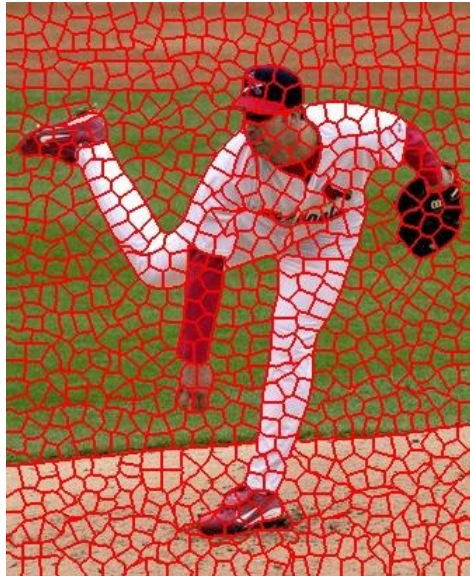
Human segmentation



Цель сегментации

- Разделите изображение на схожие "объекты"
- Сгруппировать похожие по виду пиксели для эффективности дальнейшей обработки

“superpixels”

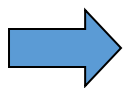


X. Ren and J. Malik. [Learning a classification model for segmentation](#). ICCV 2003.

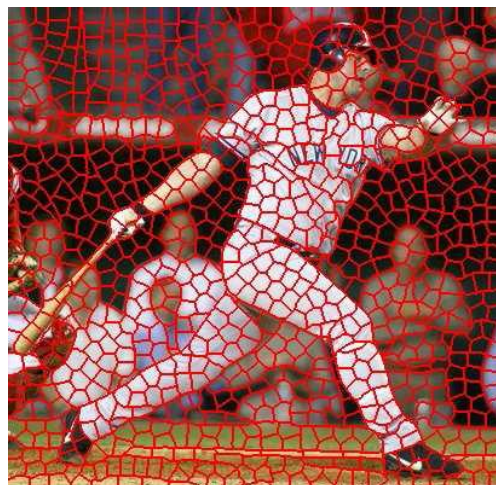
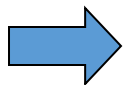
Сегментация как вспомогательная операция



Сегментация как вспомогательная операция



[Felzenszwalb and Huttenlocher 2004]



[Hoiem et al. 2005, Mori 2005]

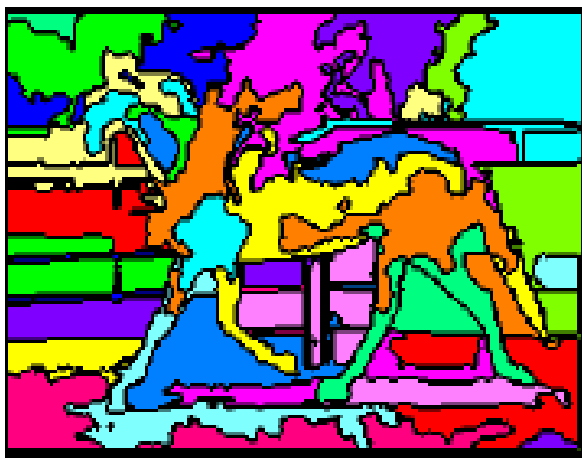
[Shi and Malik 2001]

Сегментация как вспомогательная операция

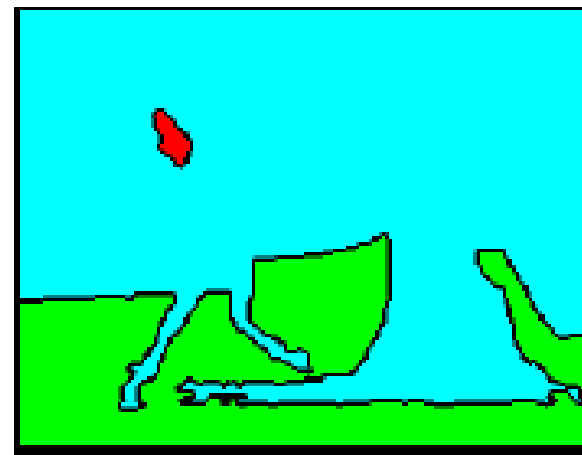


Rother et al. 2004

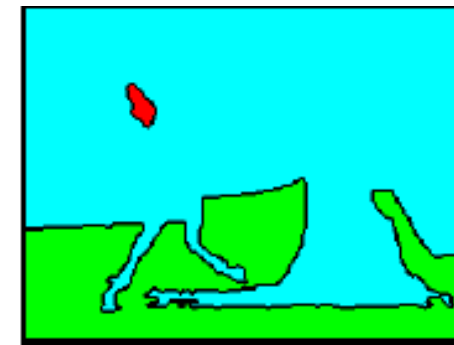
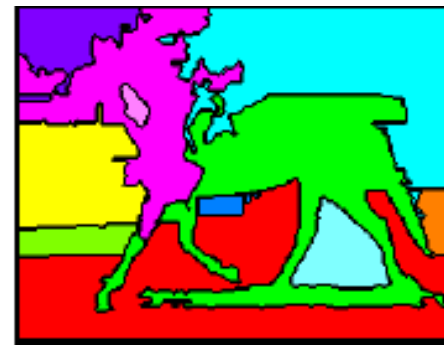
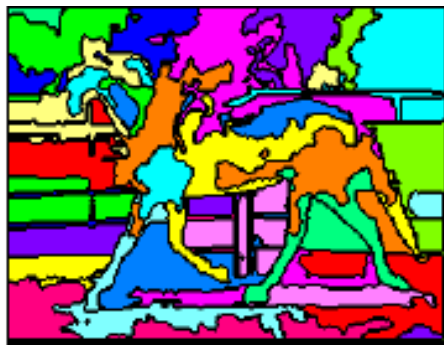
Типы сегментации



Oversegmentation



Undersegmentation

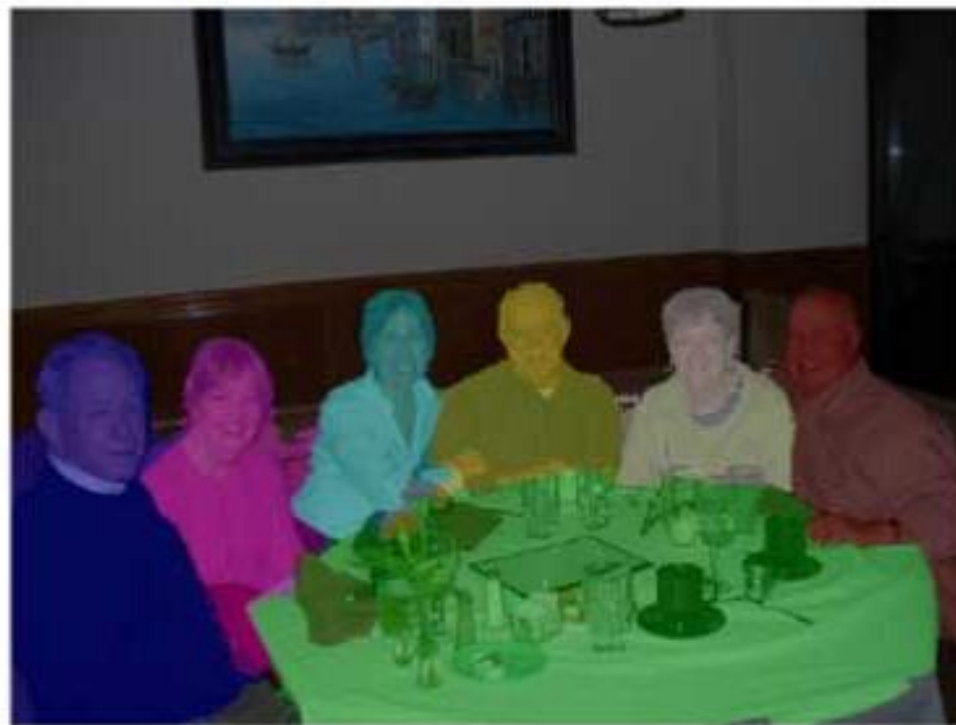


Multiple Segmentations

Типы сегментации



Semantic Segmentation



Instance Segmentation

Один из способов к сегментации - Кластеризация

Кластеризация: сгруппировать похожие точки данных и представить их одним токеном

Ключевые задачи:

- 1) Что делает похожими два пункта/изображения/патча?
- 2) Как вычислить общую группировку из парного сходства?

Зачем мы кластеризируем?

- **Суммаризация данных**

- Анализ большого объема данных
- Сжатие или шумоподавление
- Представление большого непрерывного вектора к номеру кластера

- **Подсчёт по критериям**

- Гистограммы текстуры, цвета, SIFT векторы

- **Сегментация**

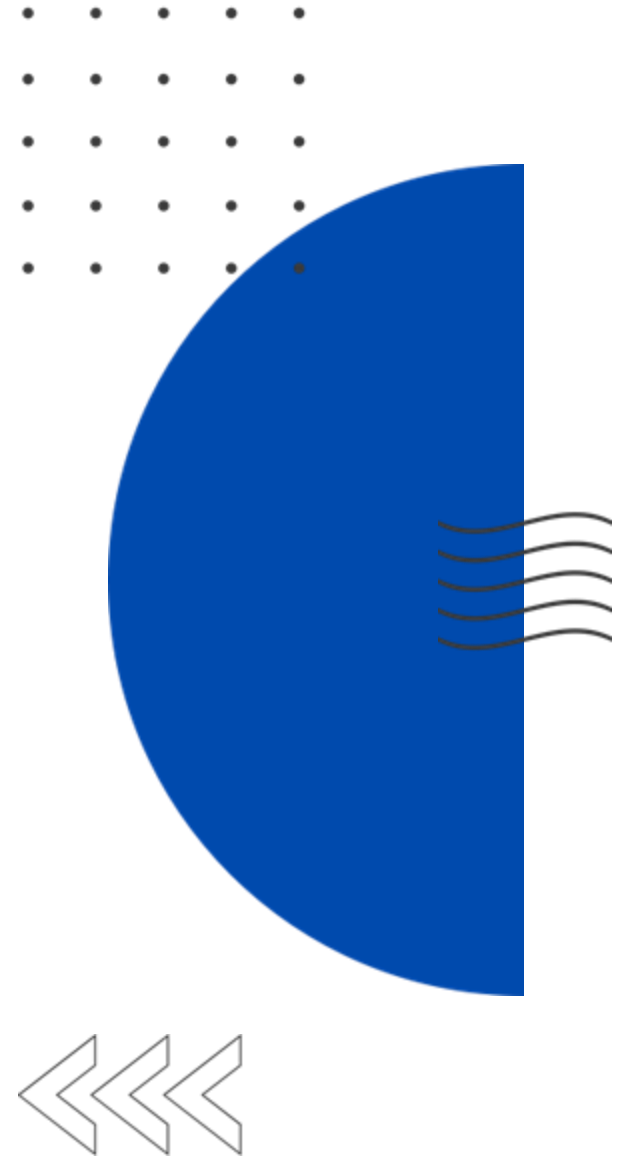
- Разделите изображение на разные регионы

- **Прогнозирование**

- Изображения в одном кластере могут иметь одинаковые метки

02

Agglomerative clustering



Желательные свойства кластерных алгоритмов

- Масштабируемость (как во времени, так и в пространстве)
- Способность работать с различными типами данных
- Минимальные требования к знаниям в области для определения входных параметров
 - Не нужно знать, сколько объектов существует или какие категории объектов будут
- Интерпретируемость и удобство использования
 - Введение ограничений и эмпирик

Кластеризация: измерение расстояния

Кластеризация - это метод обучения без учителя. Цель состоит в том, чтобы сгруппировать $x_1, \dots, x_n \in \mathbb{R}^D$ по кластерам.

Нам нужна функция парного расстояния/подобия между элементами, а иногда и желаемое количество кластеров.

Когда данные (например, изображения, объекты, документы) представлены характерными векторами, обычно используемой мерой сходства является косинусное сходство.

Пусть будут два вектора данных x, x' . Между двумя векторами есть угол θ .

Определение мер расстояния

Пусть x и x' будут двумя объектами из векторного пространства. Расстояние (подобие) между x и x' - это вещественное число, обозначаемое $\text{sim}(x, x')$.

Евклидова мера:

$$\text{sim}(x, x') = x^\top x'$$











Косинусное расстояние:

$$\begin{aligned}\text{sim}(x, x') &= \cos(\theta) \\ &= \frac{x^\top x'}{\|x\| \cdot \|x'\|} \\ &= \frac{x^\top x'}{\sqrt{x^\top x} \sqrt{x'^\top x'}}.\end{aligned}$$

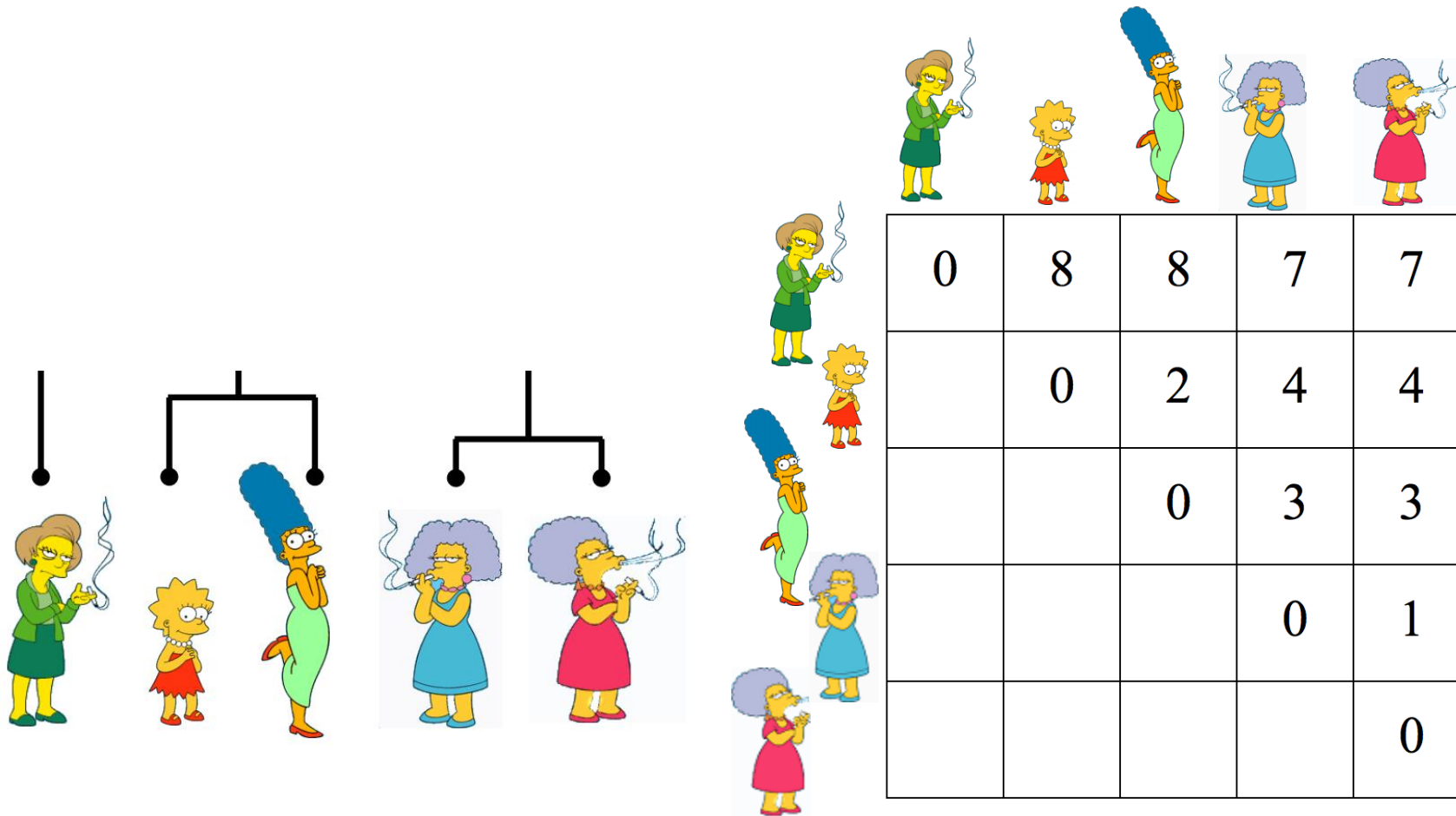
Анимированный пример

- Матрица представляет собой расстояние между двумя элементами.
- Обычно предполагается, что расстояние является обратной величиной сходства.
- Низкое расстояние означает большое сходство. ⬇ ⬇ ⬇ ⬇ ⬇

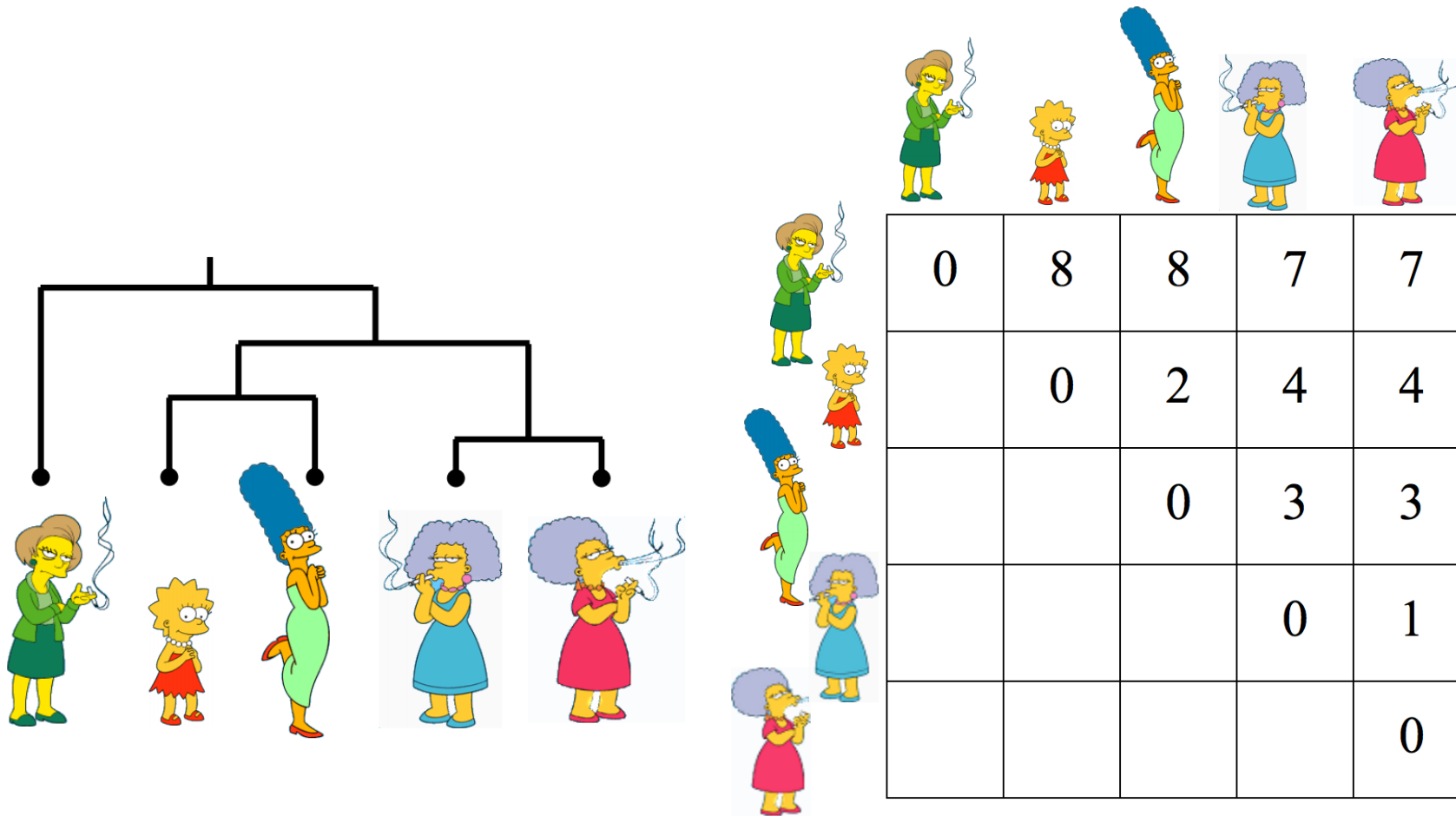


				
0	8	8	7	7
	0	2	4	4
		0	3	3
			0	1
				0
				

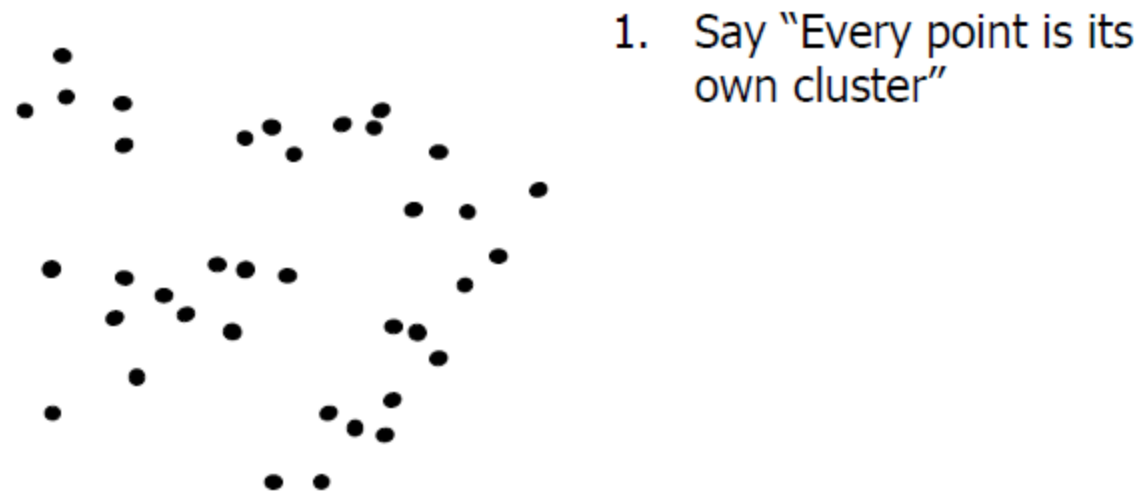
Анимированный пример



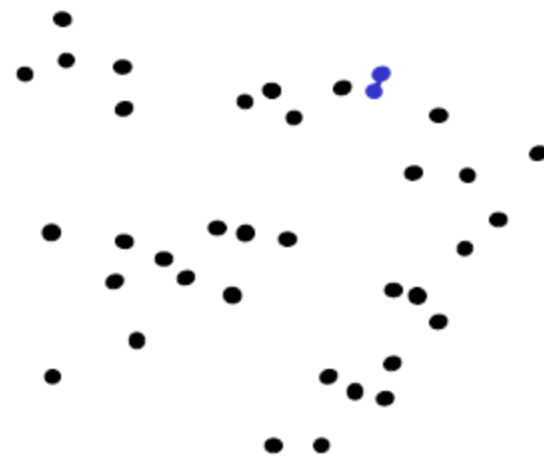
Анимированный пример



Agglomerative clustering



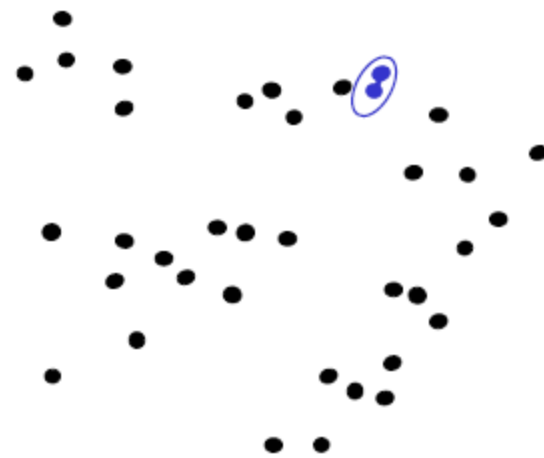
Agglomerative clustering



1. Say "Every point is its own cluster"
2. Find "most similar" pair of clusters



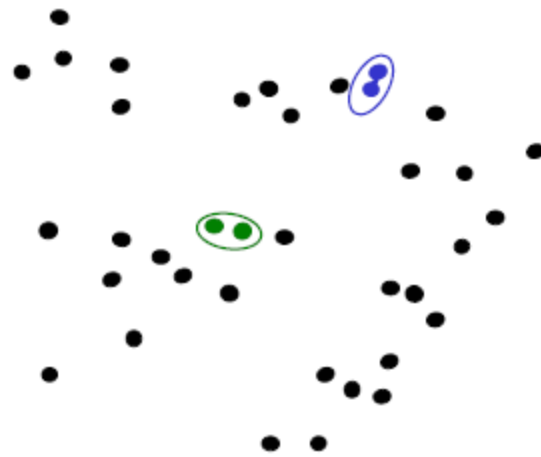
Agglomerative clustering



1. Say "Every point is its own cluster"
2. Find "most similar" pair of clusters
3. Merge it into a parent cluster



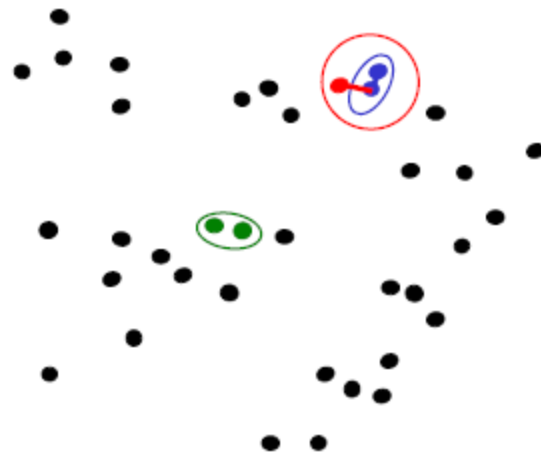
Agglomerative clustering



1. Say "Every point is its own cluster"
2. Find "most similar" pair of clusters
3. Merge it into a parent cluster
4. Repeat



Agglomerative clustering



1. Say "Every point is its own cluster"
2. Find "most similar" pair of clusters
3. Merge it into a parent cluster
4. Repeat



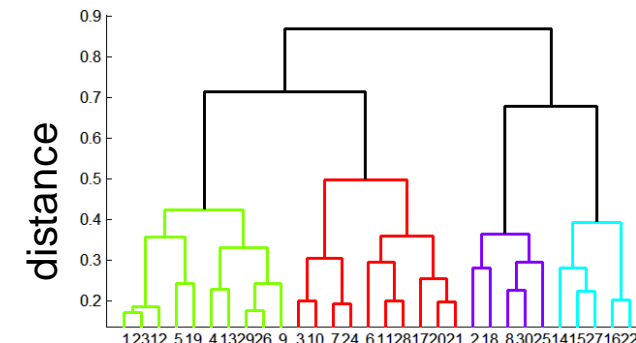
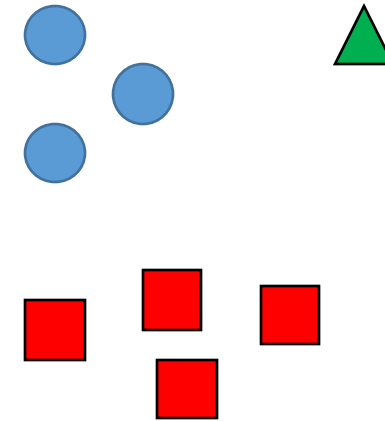
Agglomerative clustering

Как определить кластерное сходство?

- Среднее расстояние между точками,
- максимальное расстояние
- минимальная дистанция

Сколько кластеров?

- Кластеризация создает дендрограмму (дерево)
- Порог, основанный на максимальном количестве кластеров или на расстоянии между слияниями



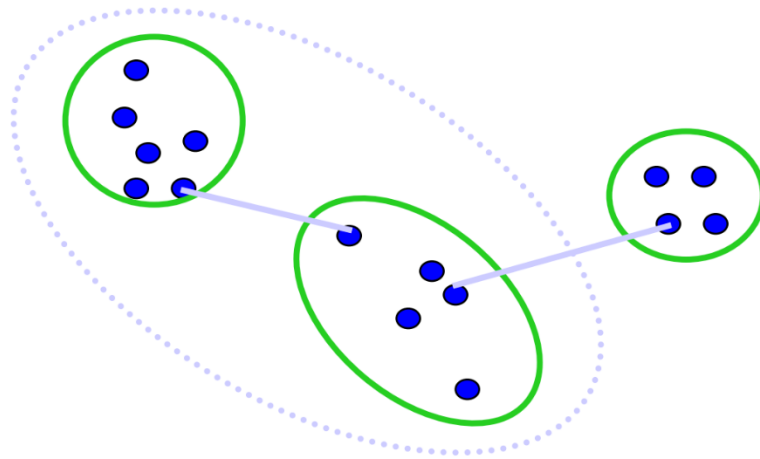
Agglomerative Hierarchical Clustering - Algorithm

1. Initially each item x_1, \dots, x_n is in its own cluster C_1, \dots, C_n .
2. Repeat until there is only one cluster left:
3. Merge the nearest clusters, say C_i and C_j .

Различные меры ближайших кластеров

Single Link

- $d(C_i, C_j) = \min_{x \in C_i, x' \in C_j} d(x, x')$. This is known as *single-linkage*. It is equivalent to the minimum spanning tree algorithm. One can set a threshold and stop clustering once the distance between clusters is above the threshold. Single-linkage tends to produce long and skinny clusters.

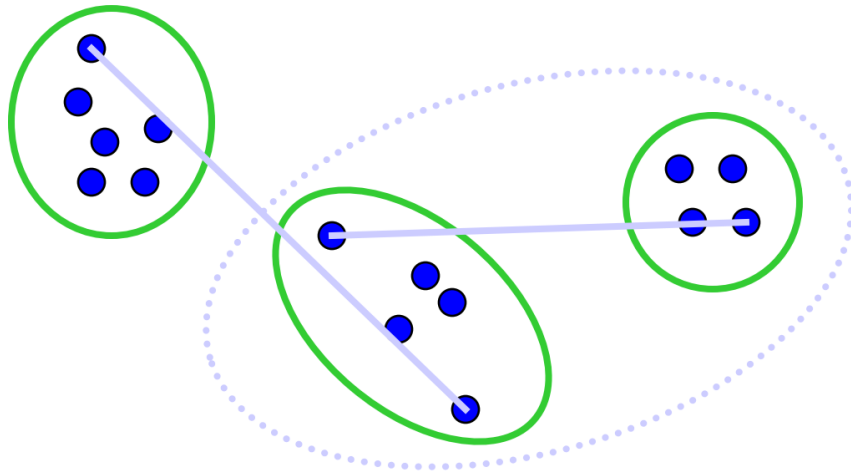


Длинные, тощие
кластеры

Различные меры ближайших кластеров

Complete Link

- $d(C_i, C_j) = \max_{x \in C_i, x' \in C_j} d(x, x')$. This is known as *complete-linkage*. Clusters tend to be compact and roughly equal in diameter.

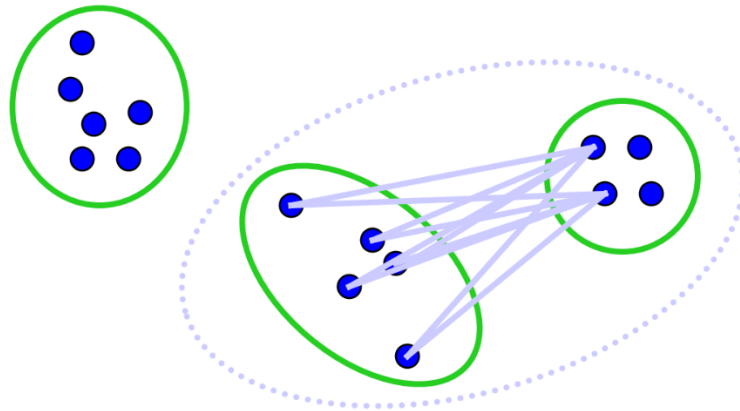


Тесные кластеры

Различные меры ближайших кластеров

Average Link

- $d(C_i, C_j) = \frac{\sum_{x \in C_i, x' \in C_j} d(x, x')}{|C_i| \cdot |C_j|}$. This is the average distance between items. Somewhere between single-linkage and complete-linkage.



Устойчивость к
шуму

Итоги: Agglomerative Clustering

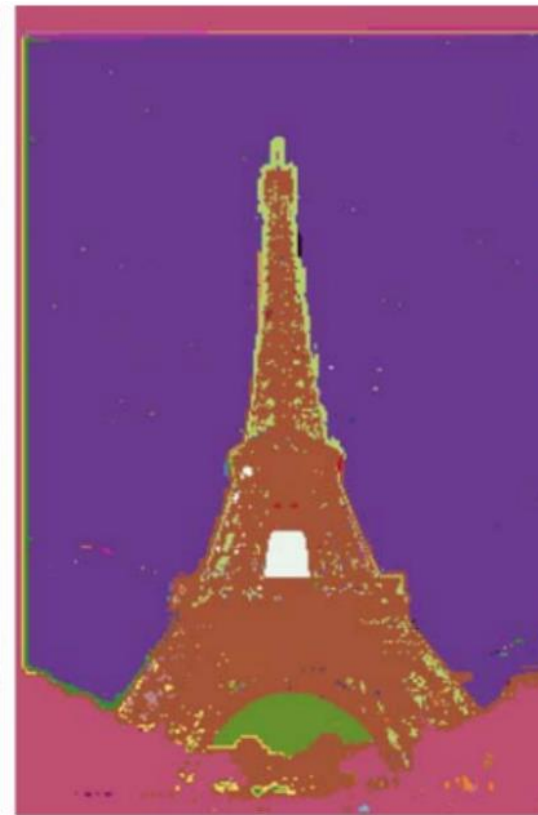
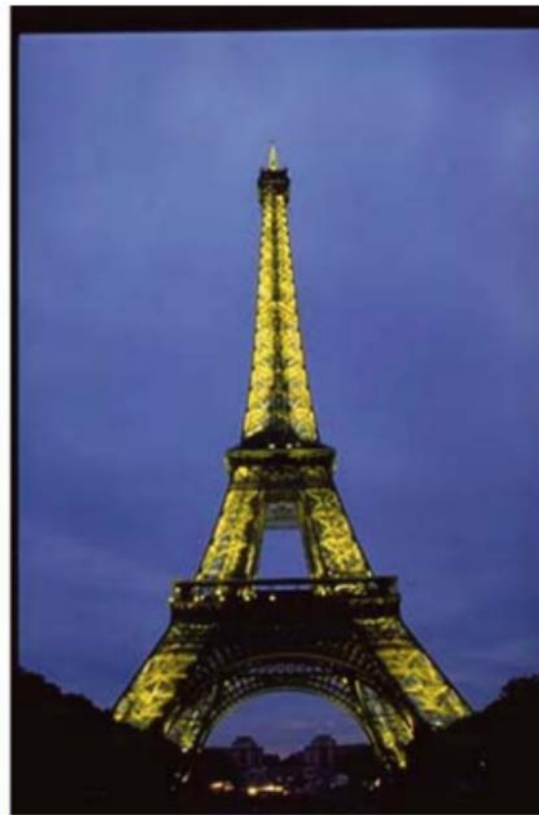
Плюсы:

- Простое в реализации, широкое применение.
- Кластеры имеют адаптивные формы.
- Обеспечивает иерархию кластеров.
- Нет необходимости заранее указывать количество кластеров.

Минусы:

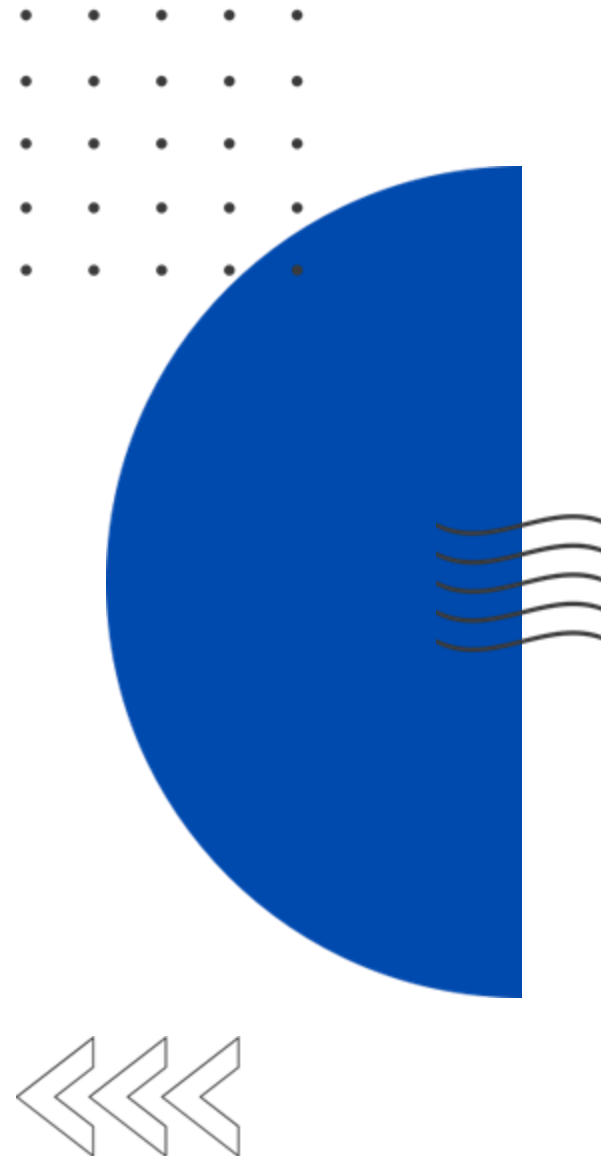
- Могут быть несбалансированные кластеры.
- Все равно придется выбирать количество кластеров или порог.
- Не очень хорошо масштабируется. Время выполнения $O(n^3)$.
- Может застрять в локальной оптике.

Результаты класетризации



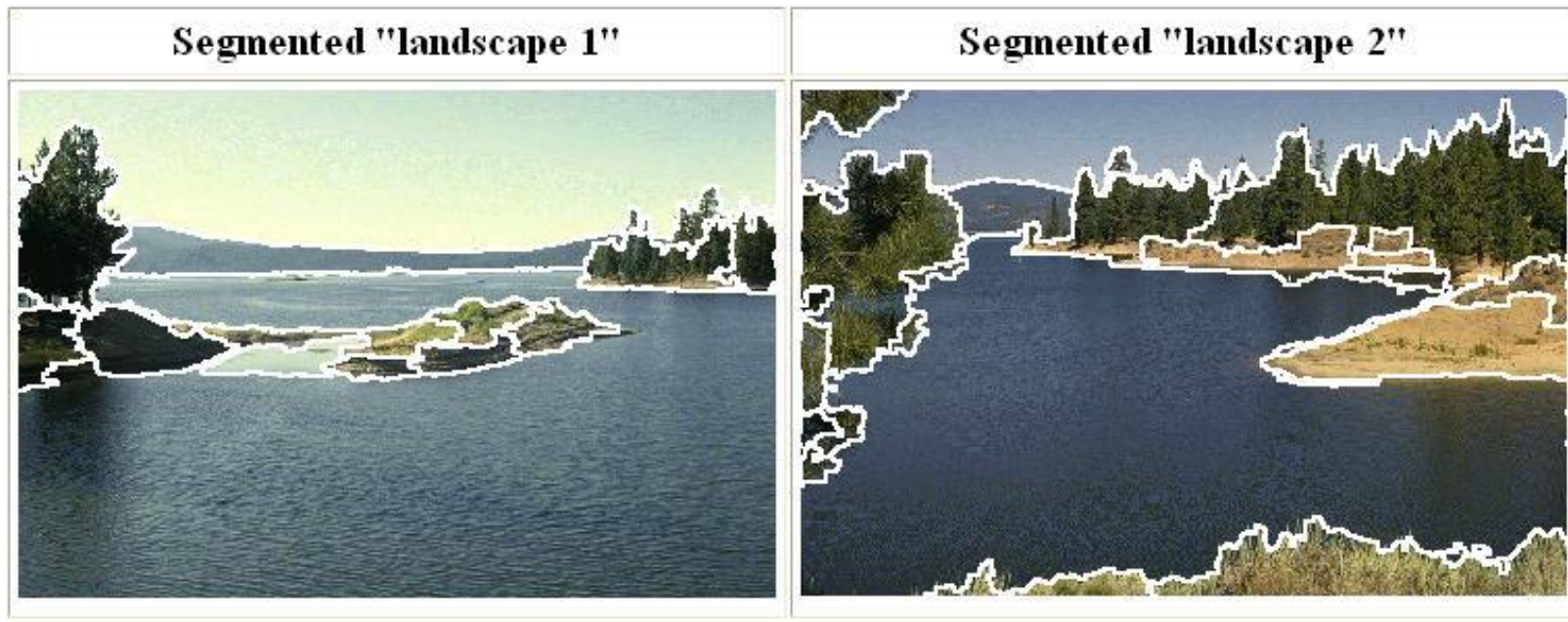
03

Mean-shift clustering



Mean-Shift Segmentation

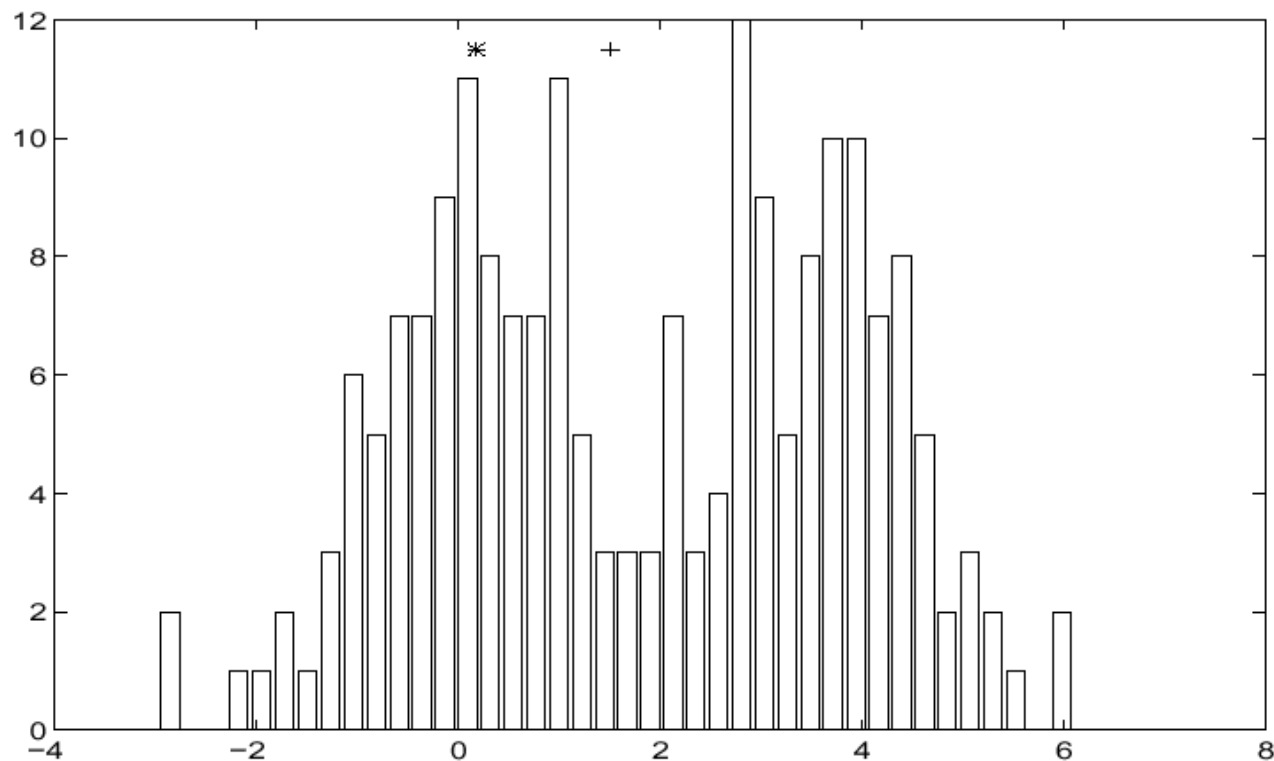
- Передовая и универсальная методика сегментации на основе кластеризации



<http://www.caip.rutgers.edu/~comanici/MSPAMI/msPamiResults.html>

D. Comaniciu and P. Meer, [Mean Shift: A Robust Approach toward Feature Space Analysis](#), PAMI 2002.

Mean-Shift Algorithm

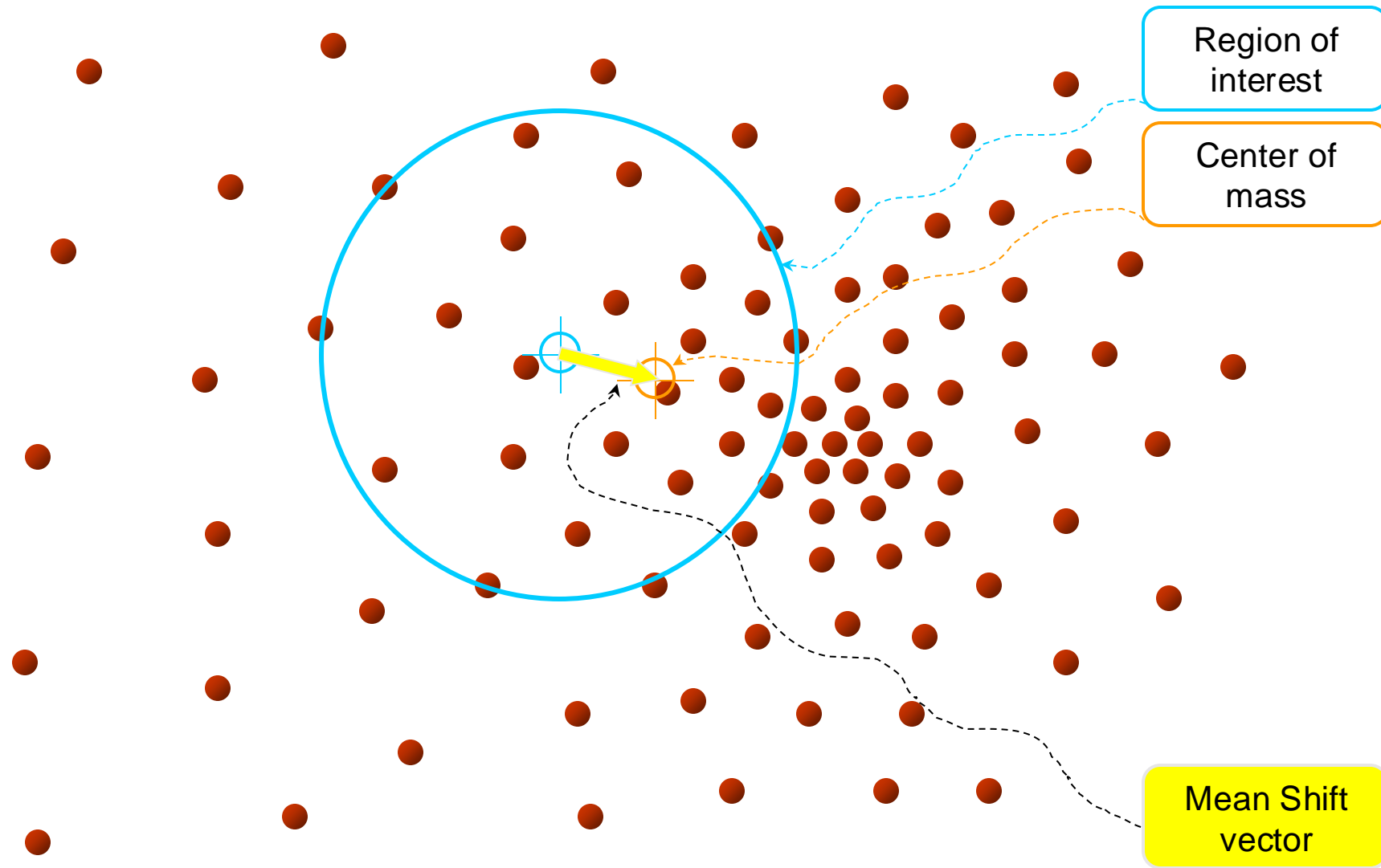


- Итеративный поиск

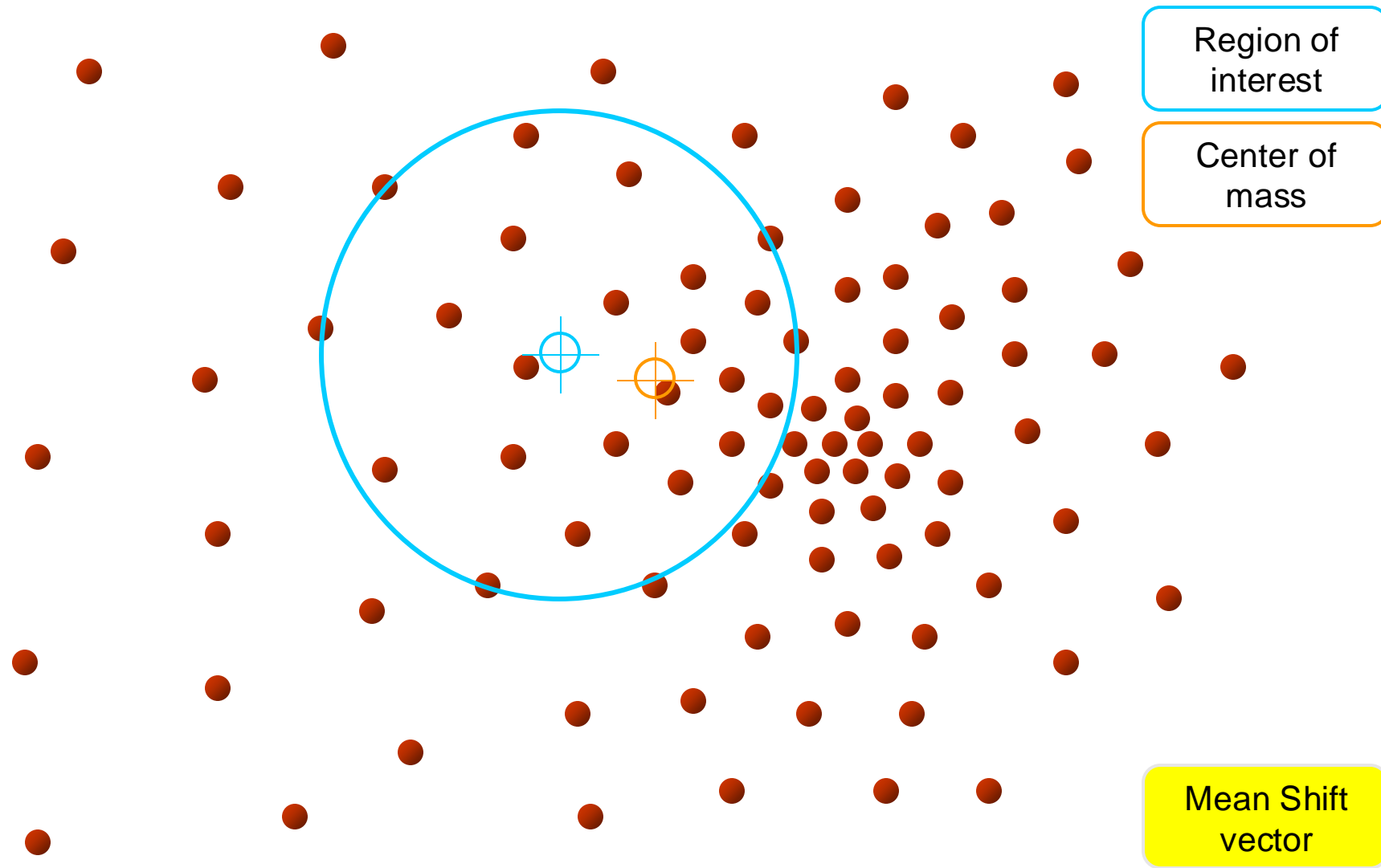
1. Инициализируйте случайный seed, и окно W
2. Вычислите центр тяжести ("средний") W:
3. Сдвинуть окно поиска на среднее
4. Повторите шаг 2 до сходимости

$$\sum_{x \in W} xH(x)$$

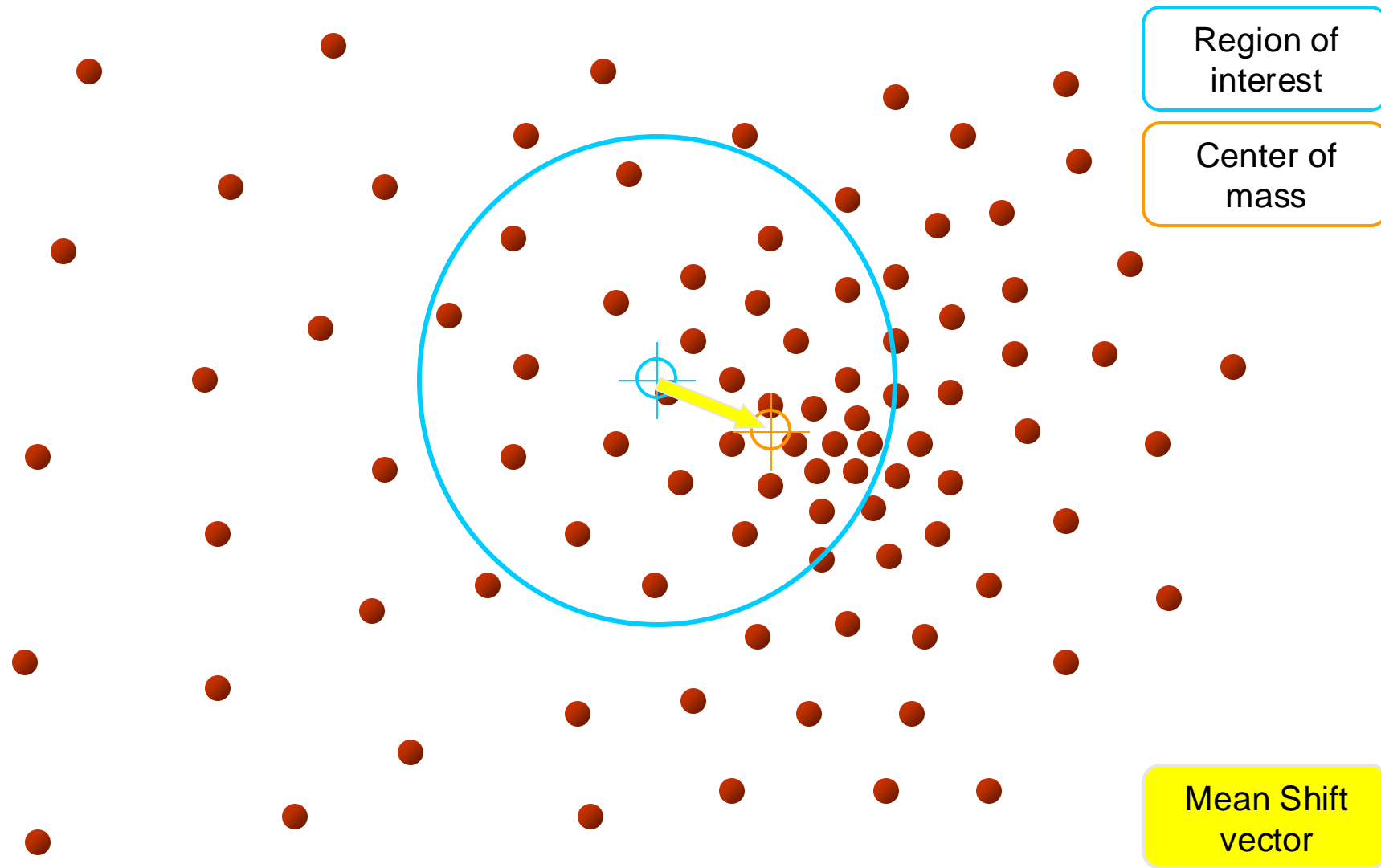
Mean-Shift



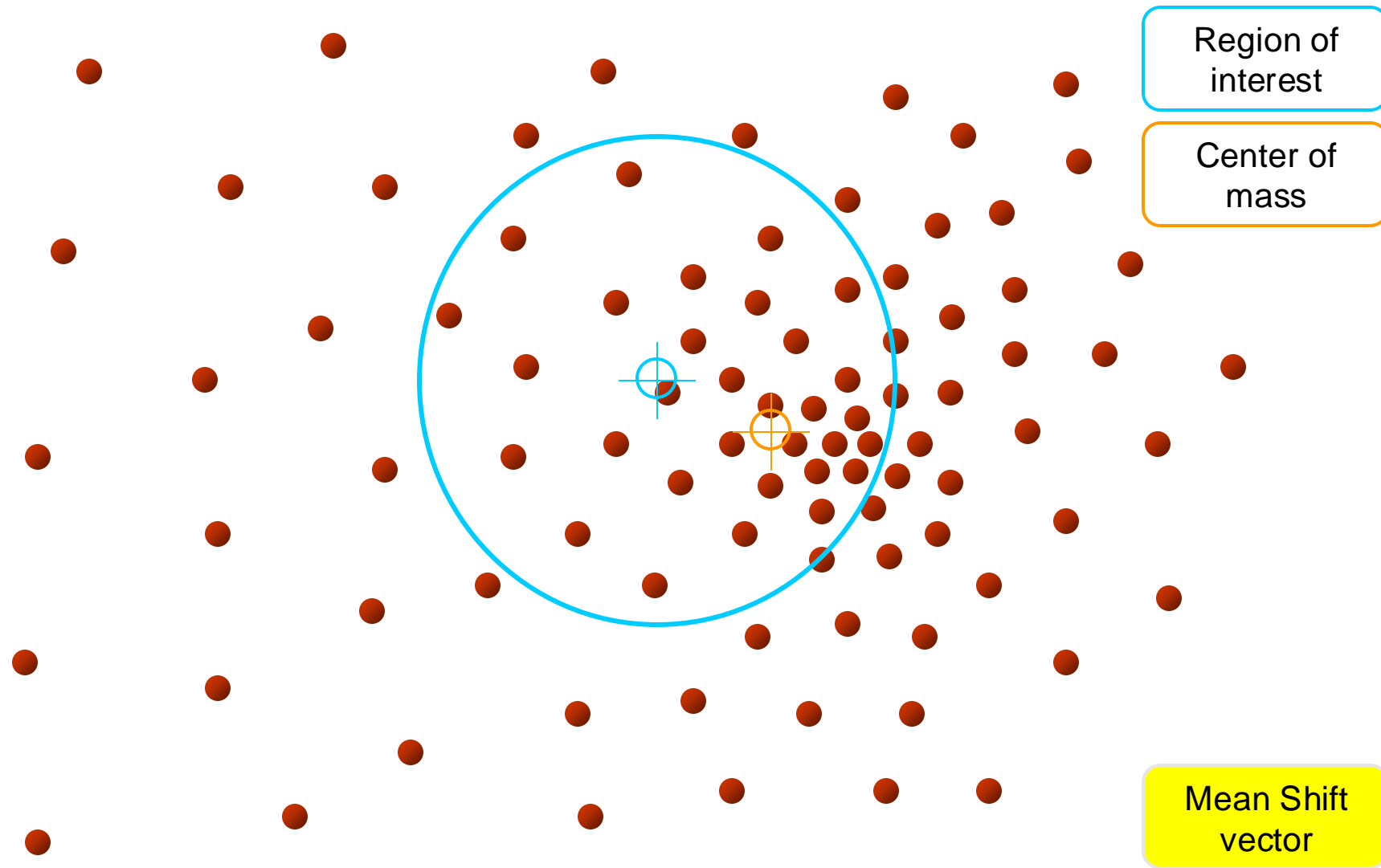
Mean-Shift



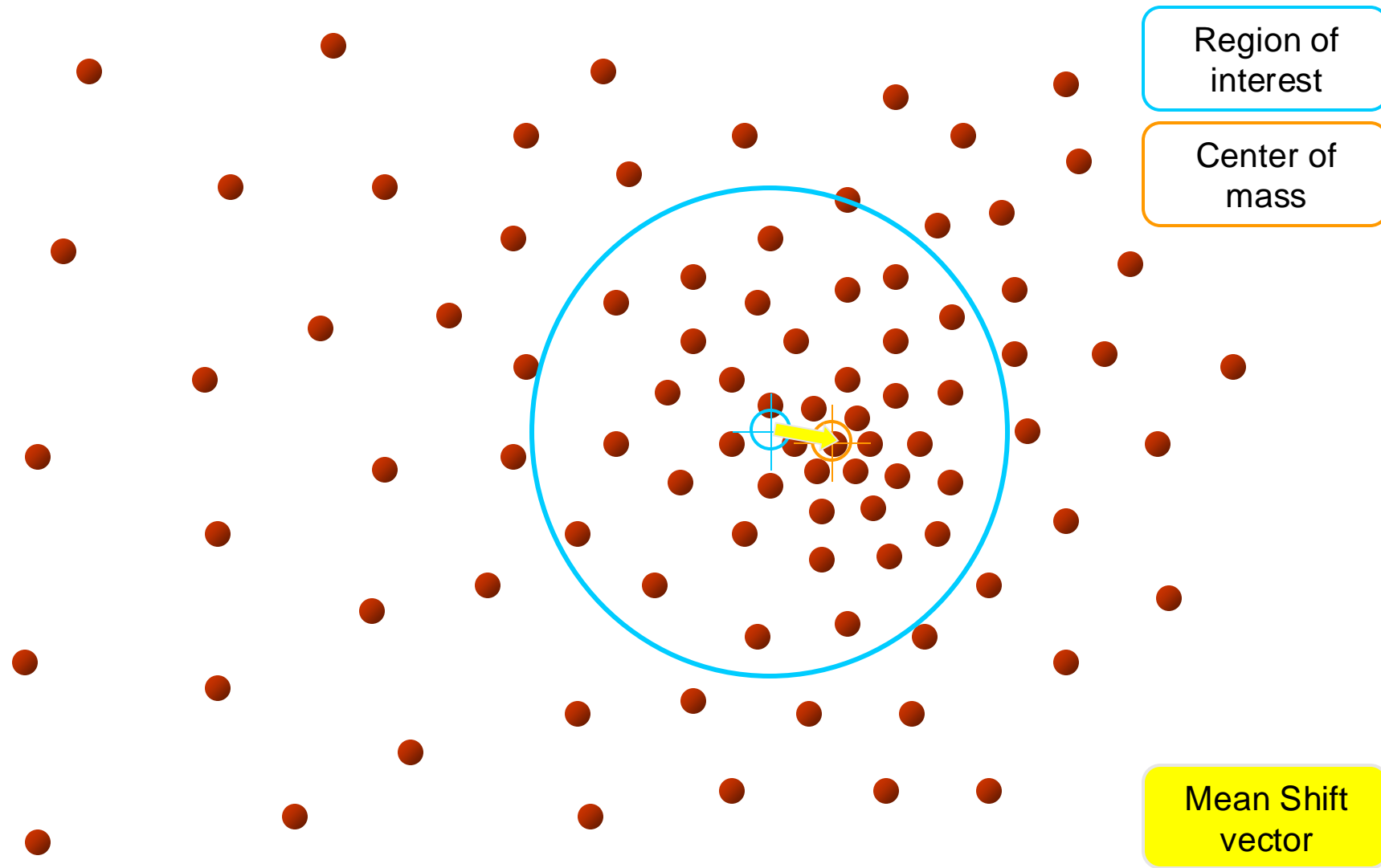
Mean-Shift



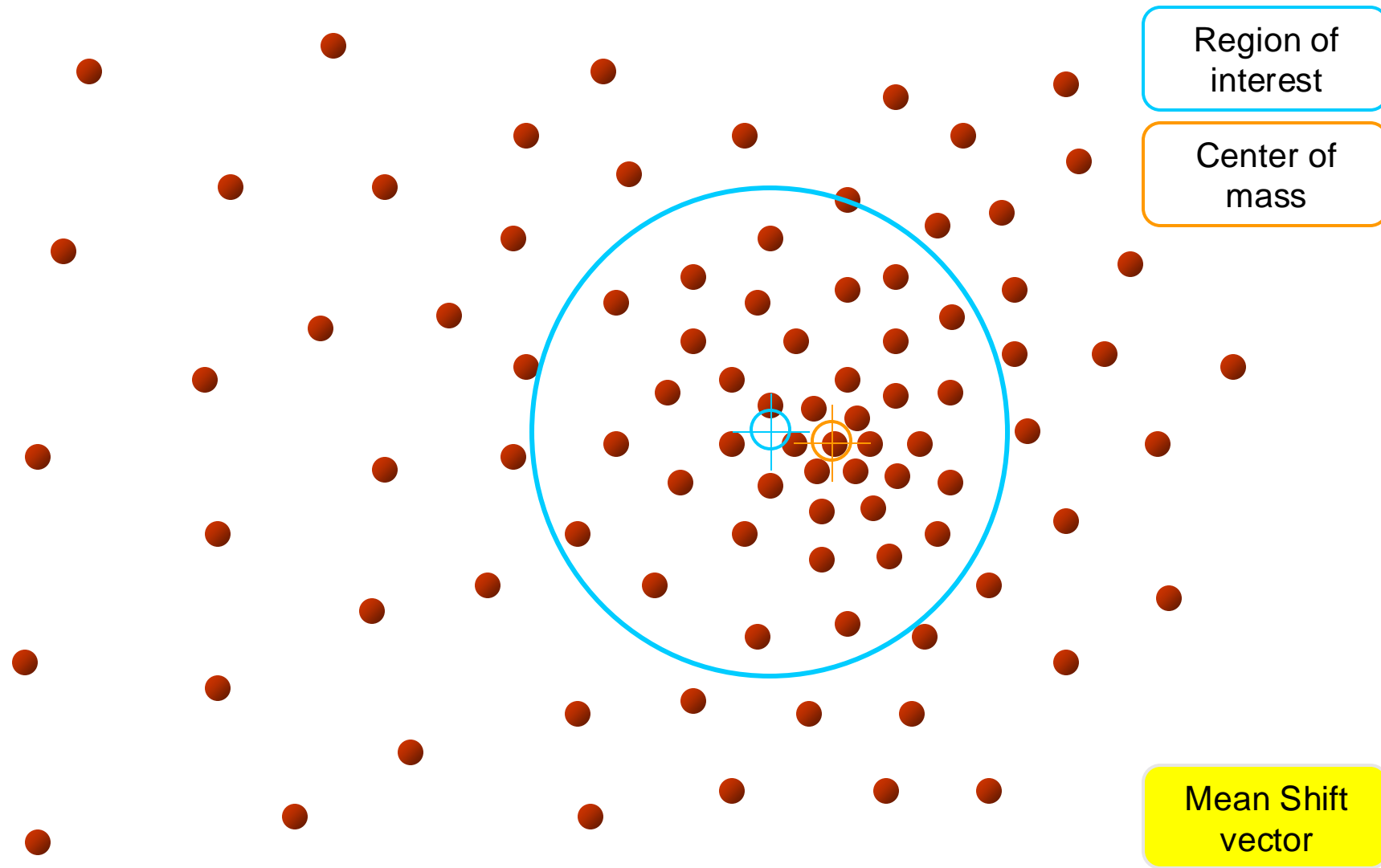
Mean-Shift



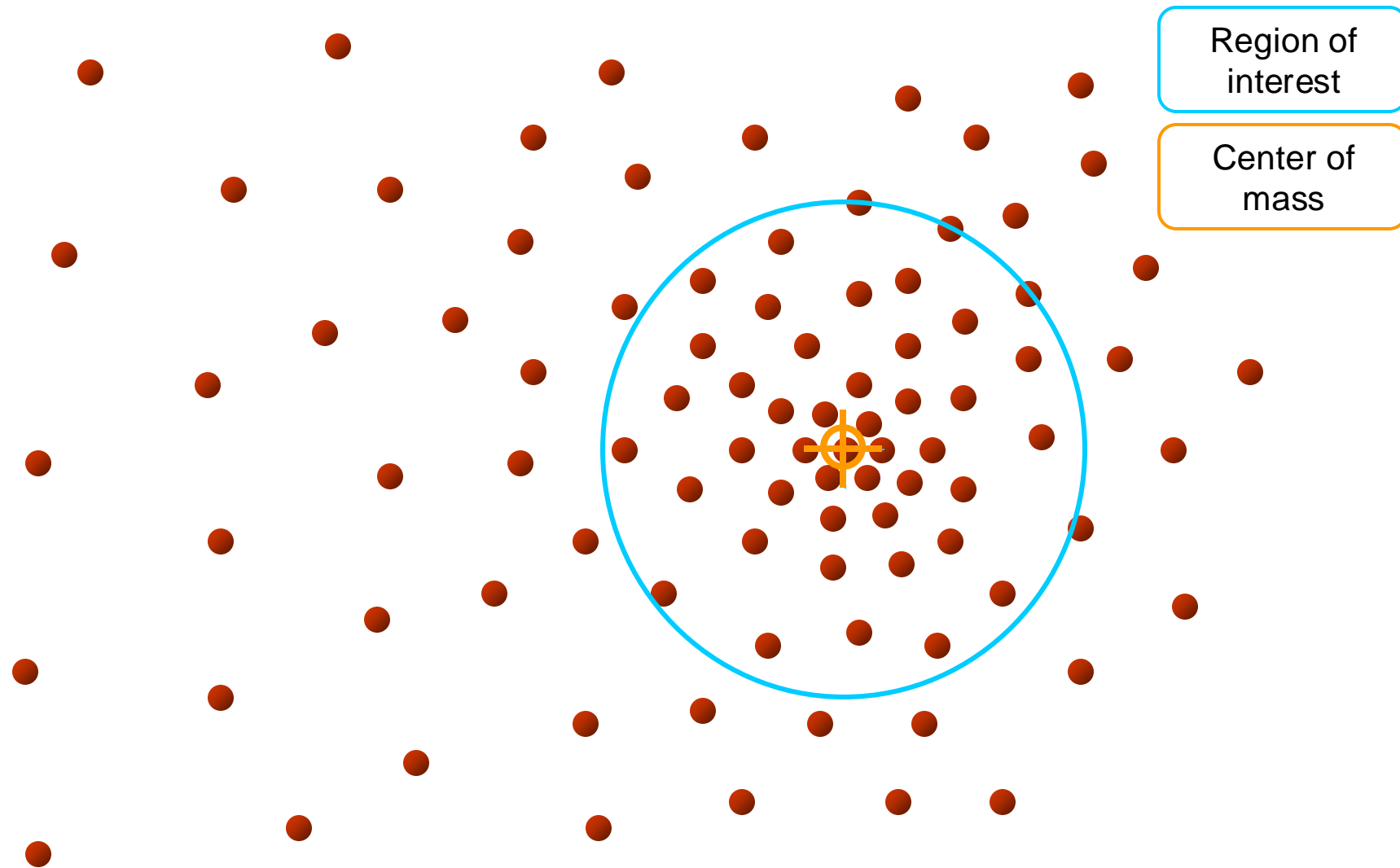
Mean-Shift



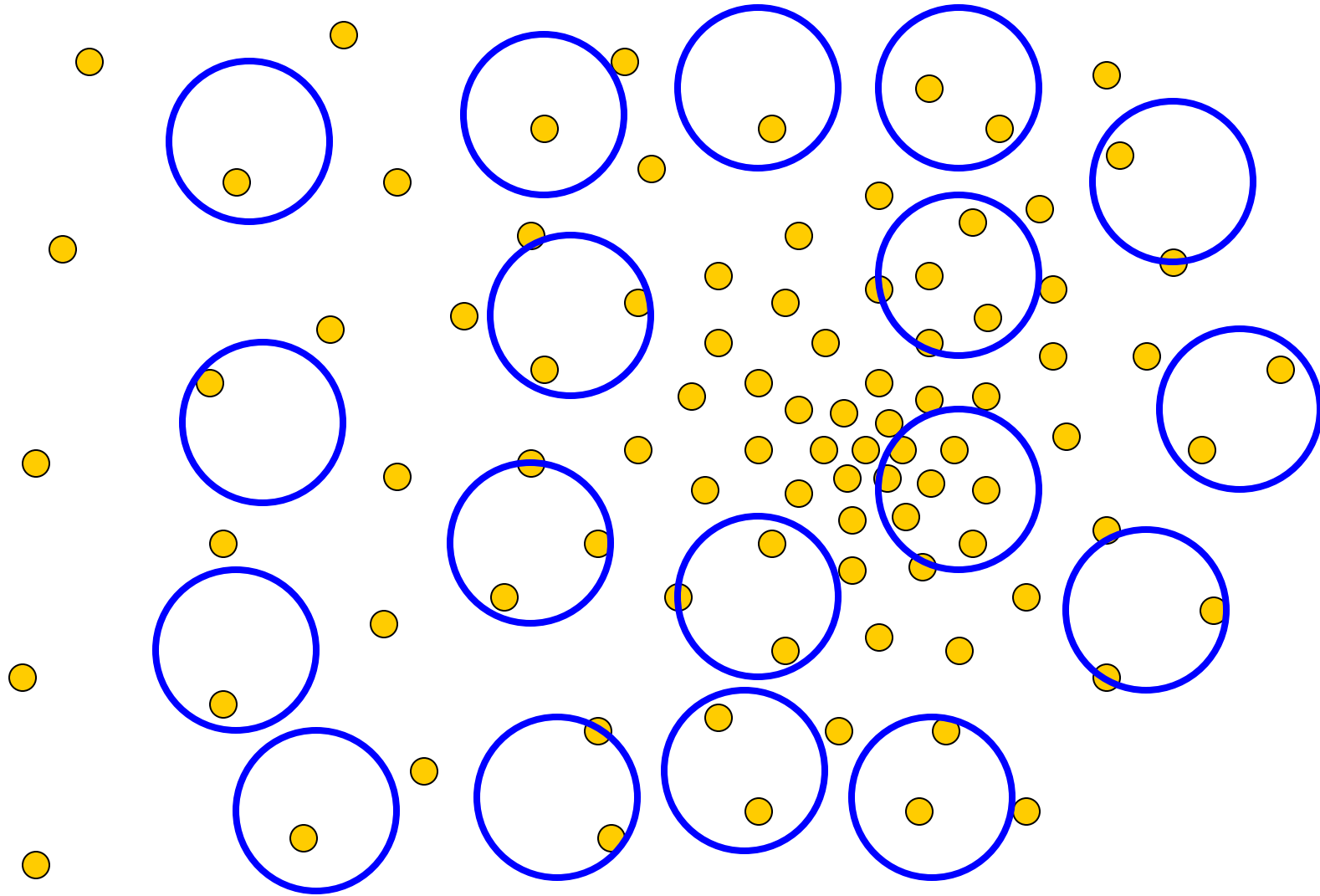
Mean-Shift



Mean-Shift



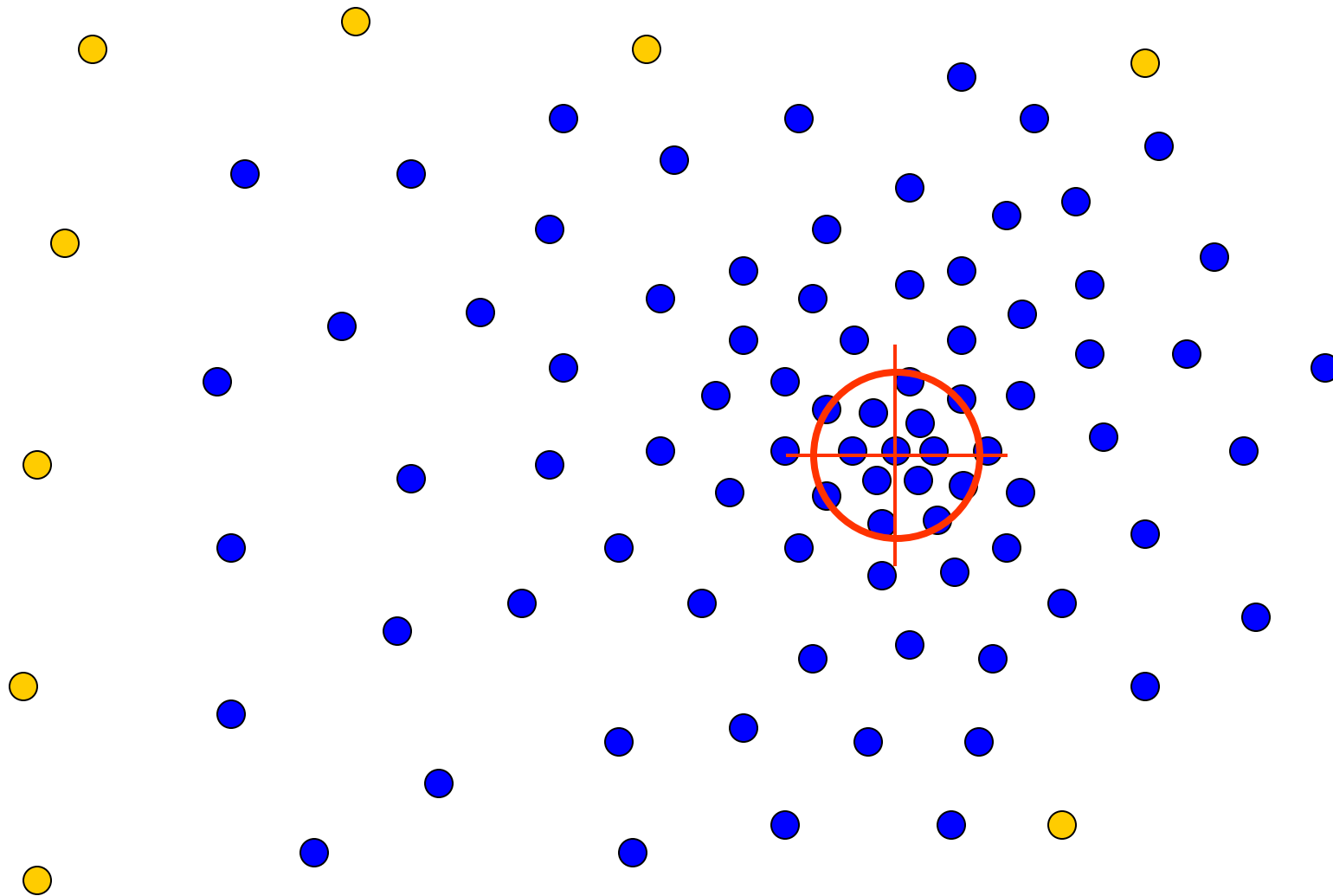
Real Modality Analysis



Tessellate the space with windows

Run the procedure in parallel

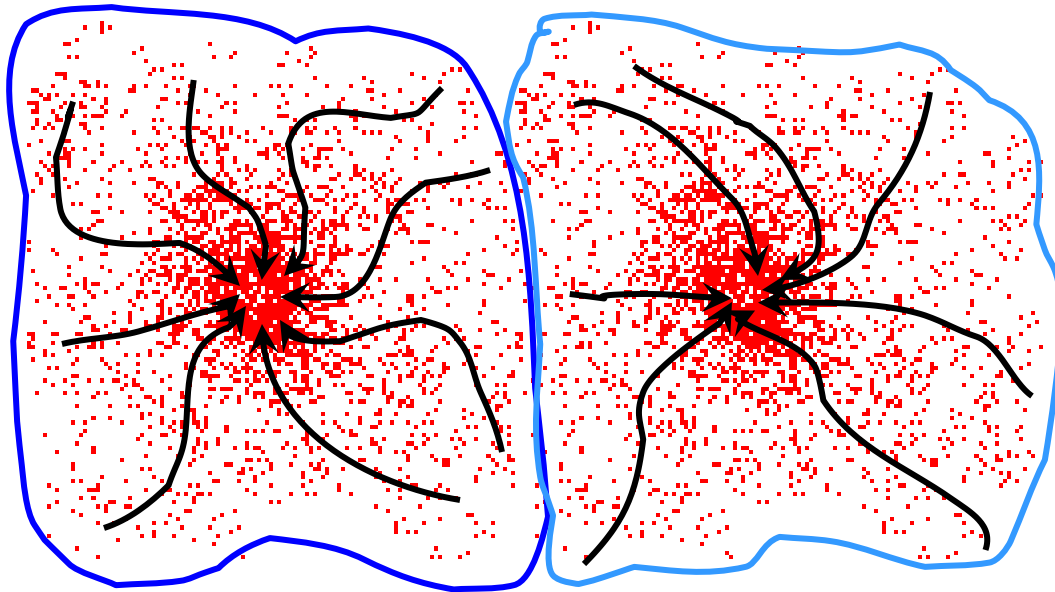
Real Modality Analysis



Голубые точки перемещались в одну сторону

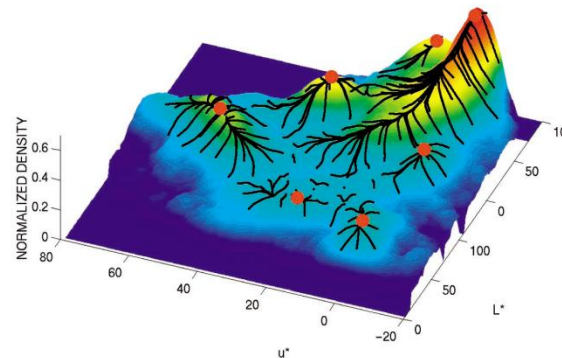
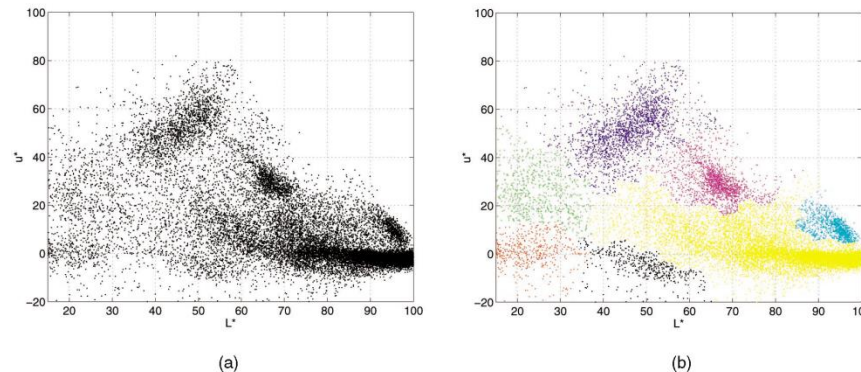
Mean-Shift Clustering

- Кластер: все точки данных в бассейне аттракционов режима
- Привлекательный бассейн: область, для которой все траектории ведут в один и тот же режим.

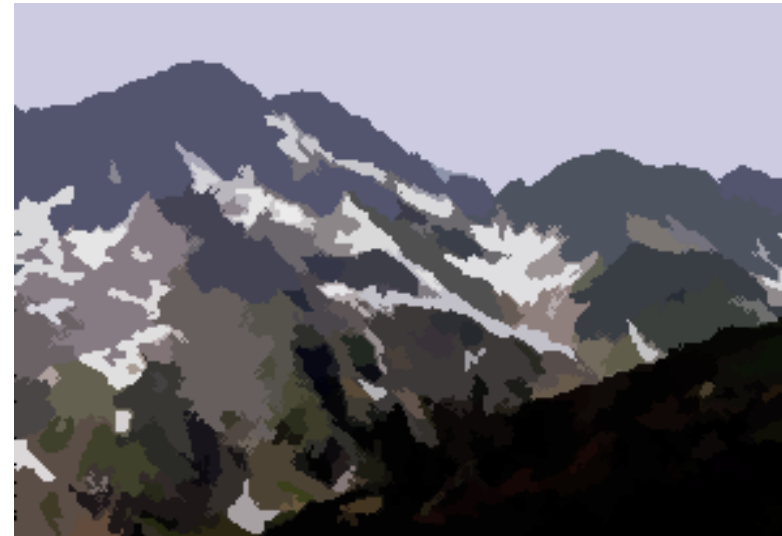


Mean-Shift Clustering/Segmentation

- Находить особенности (цвет, градиенты, текстура и т.д.).
- Инициализируйте окна в отдельных пикселях.
- Выполните среднее смещение для каждого окна до сходимости
- Объединить окна, которые заканчиваются в одном и том же "пике" или режиме.

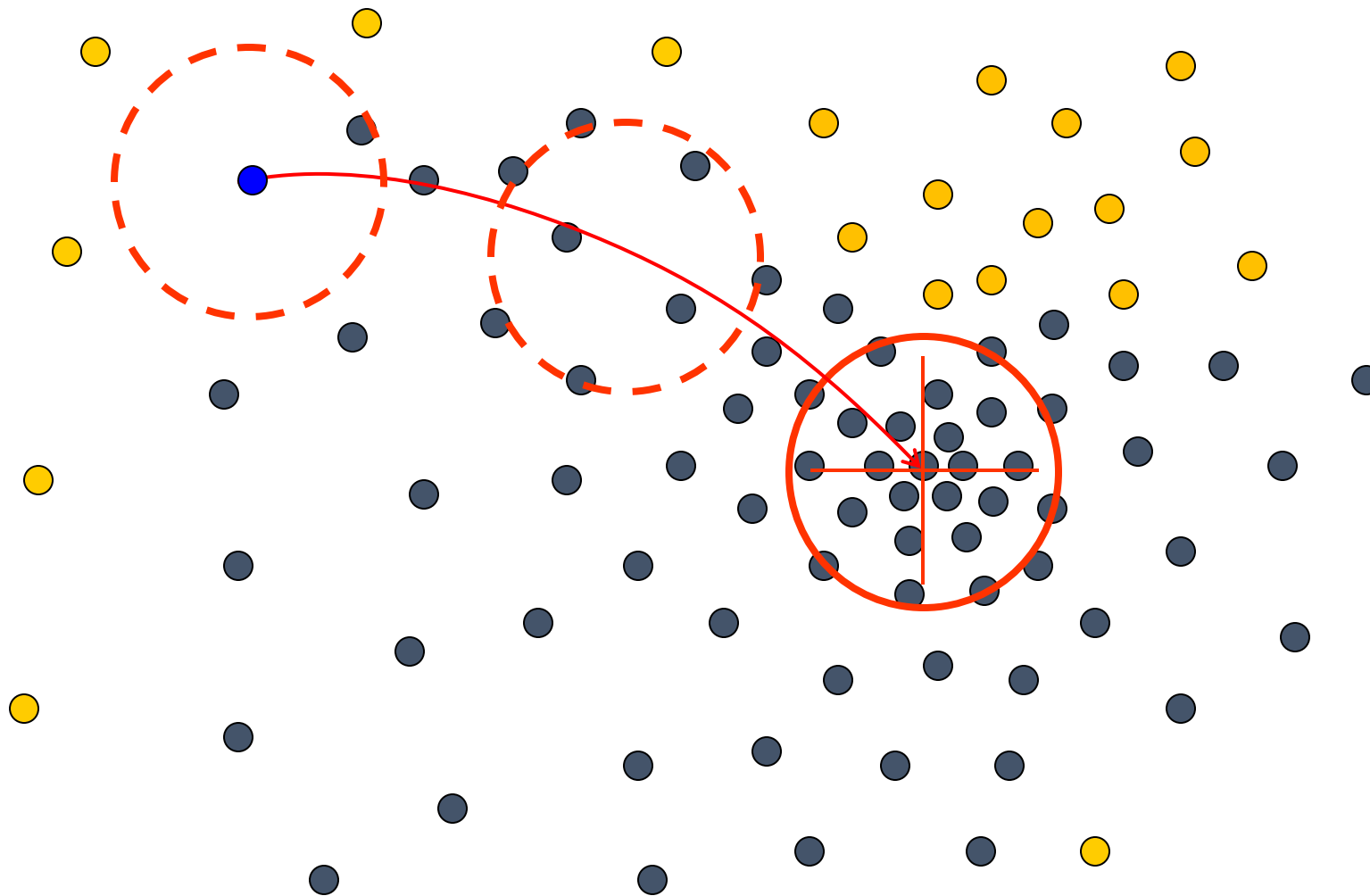


Mean-Shift Segmentation Results



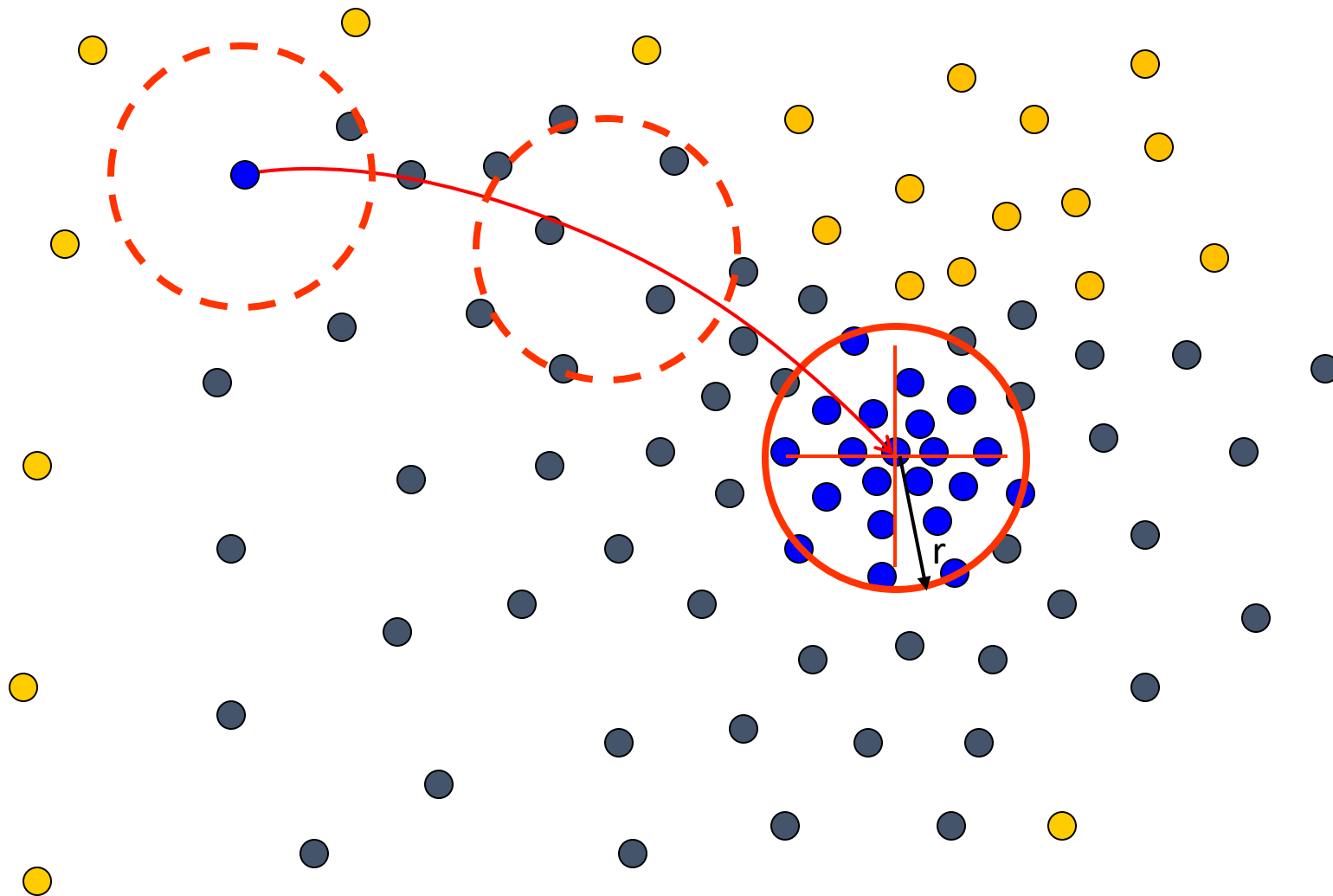
<http://www.caip.rutgers.edu/~comanici/MSPAMI/msPamiResults.html>

Проблема: Вычислительная сложность

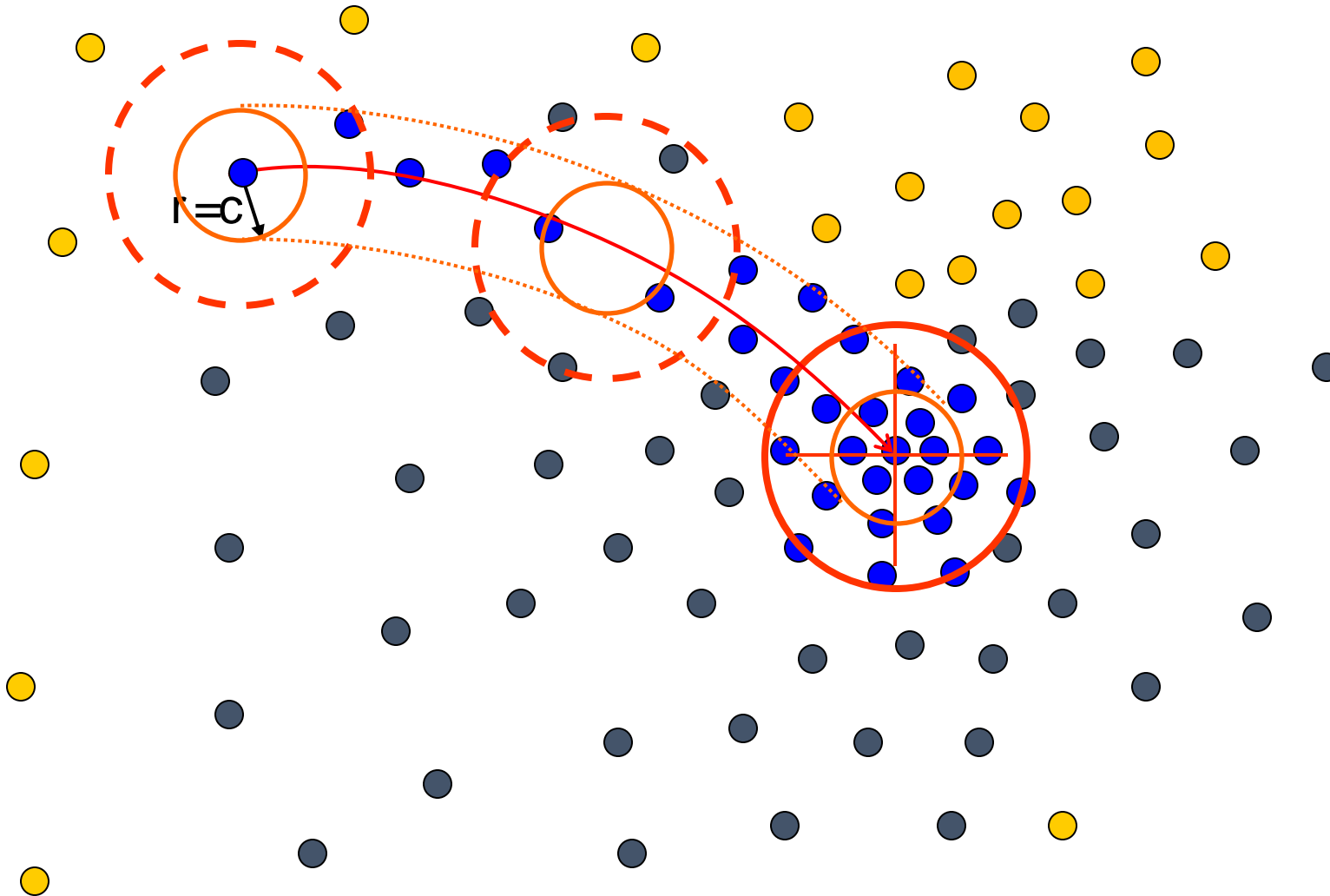


- Нужно переставить много окон...
- Многие вычисления будут излишними.

Скорость сходимости



Скорость сходимости



Присвойте всем точкам в радиусе r/c пути поиска режим -> уменьшить количество точек данных для поиска.

Технические нюансы

Given n data points $\mathbf{x}_i \in \mathbb{R}^d$, the multivariate kernel density estimate using a radially symmetric kernel¹ (e.g., Epanechnikov and Gaussian kernels), $K(\mathbf{x})$, is given by,

$$\hat{f}_K = \frac{1}{nh^d} \sum_{i=1}^n K\left(\frac{\mathbf{x} - \mathbf{x}_i}{h}\right), \quad (1)$$

where h (termed the *bandwidth* parameter) defines the radius of kernel. The radially symmetric kernel is defined as,

$$K(\mathbf{x}) = c_k k(\|\mathbf{x}\|^2), \quad (2)$$

where c_k represents a normalization constant.

Другие ядра

A kernel is a function that satisfies the following requirements :

1. $\int_{\mathbb{R}^d} \phi(x) = 1$

2. $\phi(x) \geq 0$

Some examples of kernels include :

1. Rectangular $\phi(x) = \begin{cases} 1 & a \leq x \leq b \\ 0 & \text{else} \end{cases}$

2. Gaussian $\phi(x) = e^{-\frac{x^2}{2\sigma^2}}$

3. Epanechnikov $\phi(x) = \begin{cases} \frac{3}{4}(1 - x^2) & \text{if } |x| \leq 1 \\ 0 & \text{else} \end{cases}$

Технические нюансы

Взять производную: $\hat{f}_K = \frac{1}{nh^d} \sum_{i=1}^n K\left(\frac{\mathbf{x} - \mathbf{x}_i}{h}\right)$

$$\nabla \hat{f}(\mathbf{x}) = \underbrace{\frac{2c_{k,d}}{nh^{d+2}} \left[\sum_{i=1}^n g\left(\left\|\frac{\mathbf{x} - \mathbf{x}_i}{h}\right\|^2\right) \right]}_{\text{term 1}} \underbrace{\left[\frac{\sum_{i=1}^n \mathbf{x}_i g\left(\left\|\frac{\mathbf{x} - \mathbf{x}_i}{h}\right\|^2\right)}{\sum_{i=1}^n g\left(\left\|\frac{\mathbf{x} - \mathbf{x}_i}{h}\right\|^2\right)} - \mathbf{x} \right]}_{\text{term 2}}, \quad (3)$$

where $g(x) = -k'(x)$ denotes the derivative of the selected kernel profile.

- Term 1: это пропорционально оценке плотности при \mathbf{x} (аналогично уравнению 1 - два слайда назад).
- Term 2: это вектор среднего сдвига, который указывает в направлении максимальной плотности.

Comaniciu & Meer, 2002

Технические нюансы

Наконец, процедура среднего сдвига от заданной точки \mathbf{x}^t :

1. Компьютер средний вектор сдвига \mathbf{m} :

$$\mathbf{m}_{h,G}(\mathbf{x}) = \frac{\sum_{i=1}^n \mathbf{x}_i g\left(\left\|\frac{\mathbf{x}-\mathbf{x}_i}{h}\right\|^2\right)}{\sum_{i=1}^n g\left(\left\|\frac{\mathbf{x}-\mathbf{x}_i}{h}\right\|^2\right)} - \mathbf{x},$$

2. Переведите окно плотности:

$$\mathbf{x}_i^{t+1} = \mathbf{x}_i^t + \mathbf{m}(\mathbf{x}_i^t).$$

3. Итерируйте шаги 1 и 2 до сходимости.

$$\nabla f(\mathbf{x}_i) = 0.$$

Итоги: Mean-Shift

- **Плюсы:**

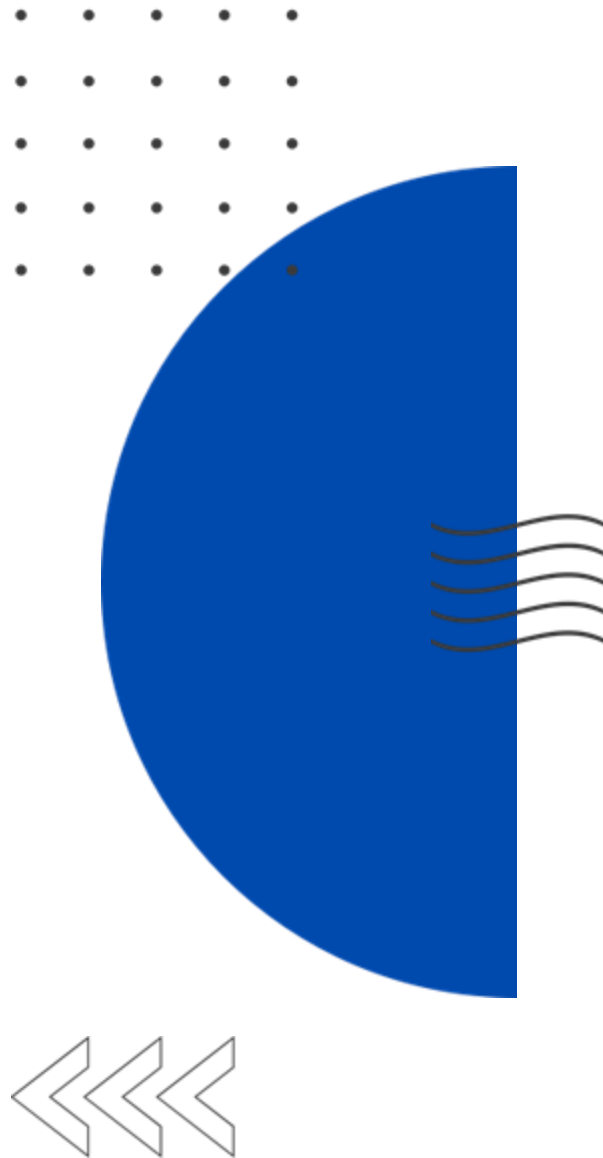
- Общий, независимый от применения инструмент
- Не содержит моделей, не принимает никакой предшествующей формы (сферической, эллиптической и т.д.) на кластеры данных
- Только один параметр (размер окна h)
 - h имеет физическое значение (в отличие от k -средних)
- Находит переменное количество режимов
- Надежен на прорыв

- **Минусы:**

- Выход зависит от размера окна
- Выбор размера окна (полосы пропускания) не тривиален.
- Вычислительно (относительно) дорого (~ 2 с/изображение)
- Плохо масштабируется в зависимости от размера художественного пространства.


04

Метрики



Segmentation: Metric

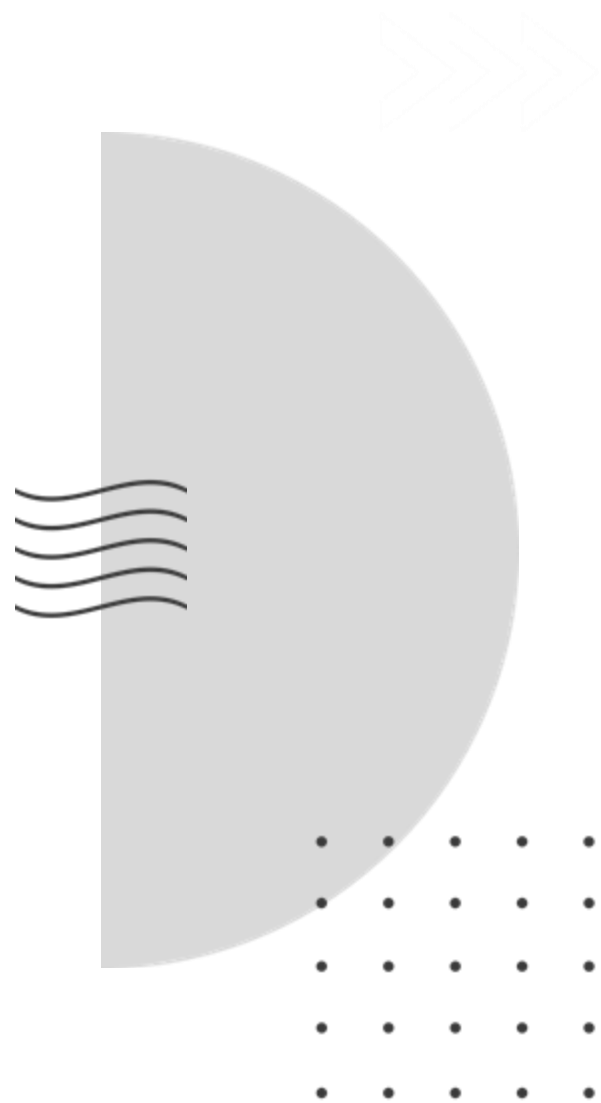
Intersection over Union (IOU) or Jaccard Index = $\frac{\text{Area of Overlap}}{\text{Area of Union}}$



Чаще всего используют
Dice - особенно в
медицинских снимках
и Jaccard (IoU)

Table 1. The three similarity coefficients

Similarity Coefficient (X,Y)	Actual Formula
Dice Coefficient	$2 \frac{ X \cap Y }{ X + Y }$
Cosine Coefficient	$\frac{ X \cap Y }{ X ^{1/2} \cdot Y ^{1/2}}$
Jaccard Coefficient	$\frac{ X \cap Y }{ X + Y - X \cap Y }$



Место для ваших
вопросов