Соловьев С.Ю.

Постановки задач современной информатики

park.glossary.ru/modern/

Задачи распознавания образов

2015 – 2022

Распознавание образов – раздел информатики, в котором разрабатываются методы основанной на прецедентах классификации объектов-образов по нескольким категориям-классам.

Напоминание: **З а д а ч а**

Дано	Известно
Исходные данные	***Свойства исх. данных***

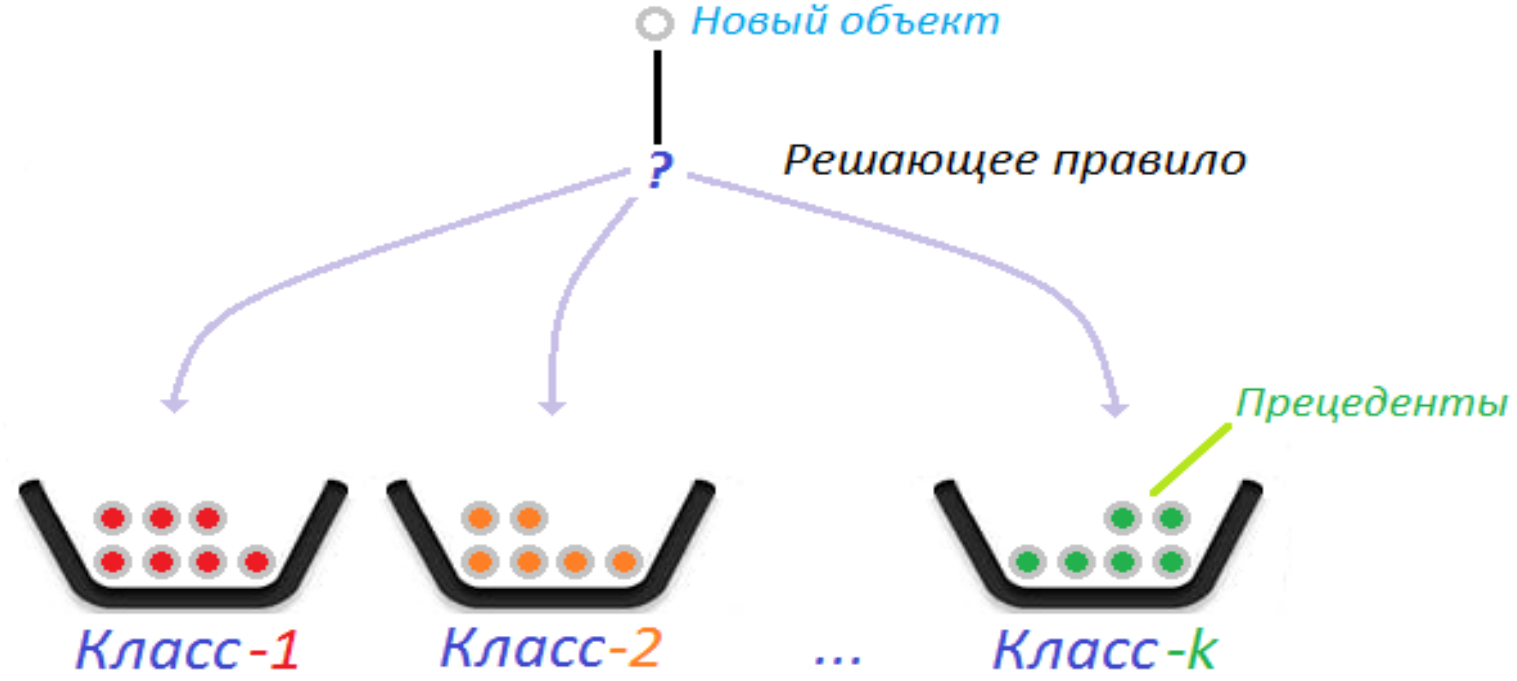
Алгоритм / Метод / Способ / Схема

Требуется	такое, что
Результирующие данные	***Свойства рез. данных***

Контекст распознавания образов

Объект ⟷ в.о.с. → $(x_1, \ldots, x_n)$ – вектор признаков

→ нек. количественное измерение объекта

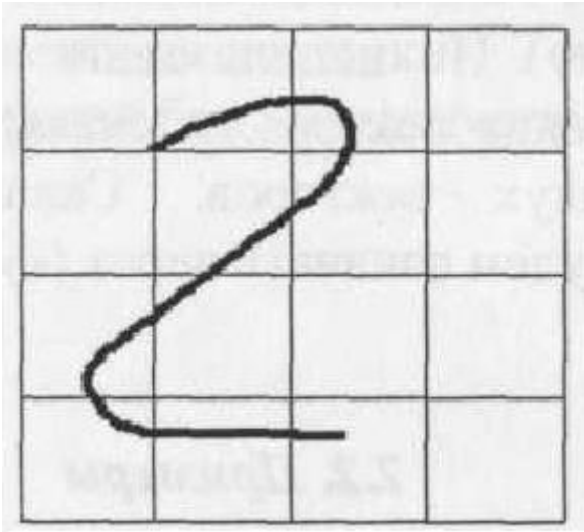


Признаки

Люди: Вес, Рост, Доход

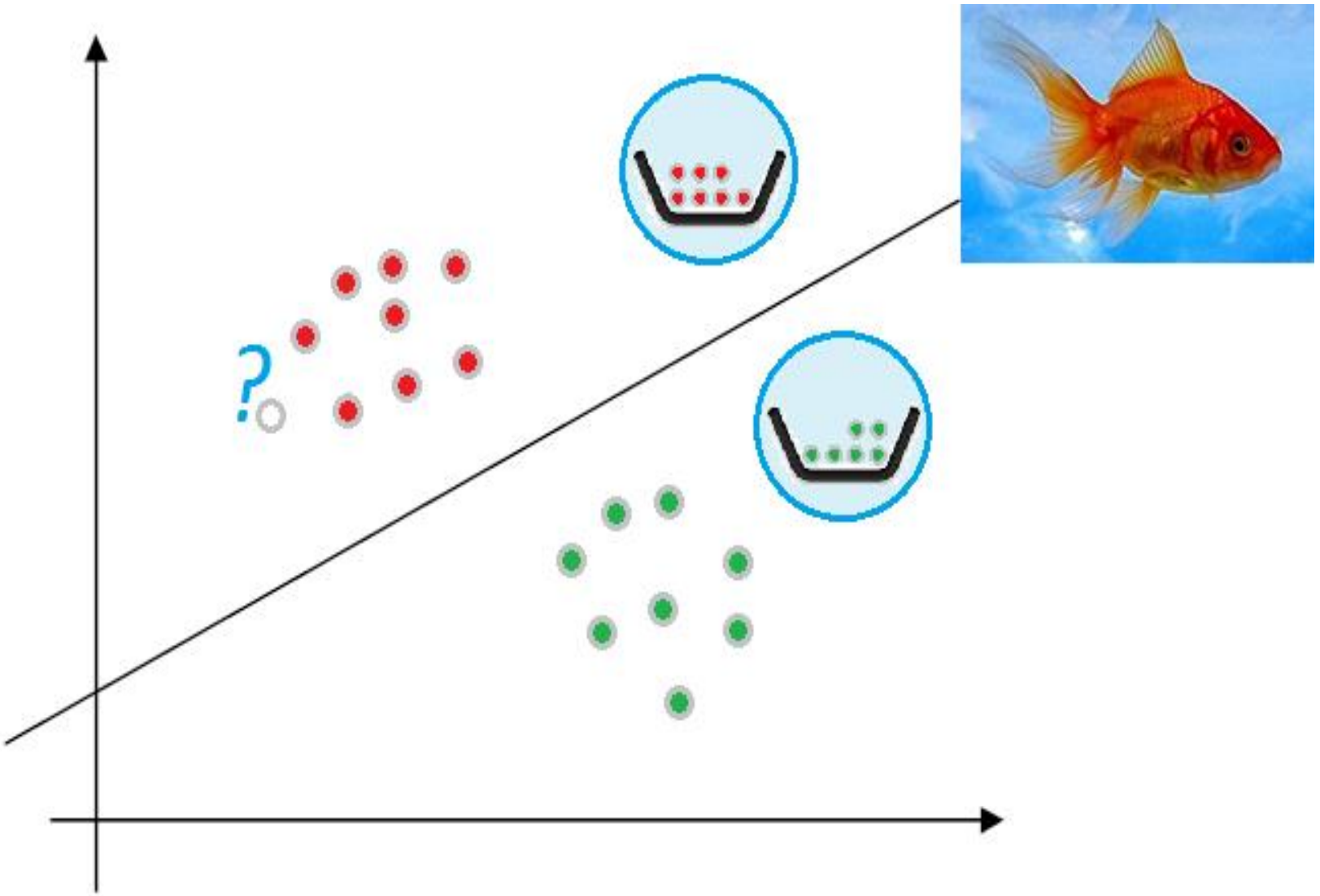
Страны: ВВП, Площадь

Изображения:

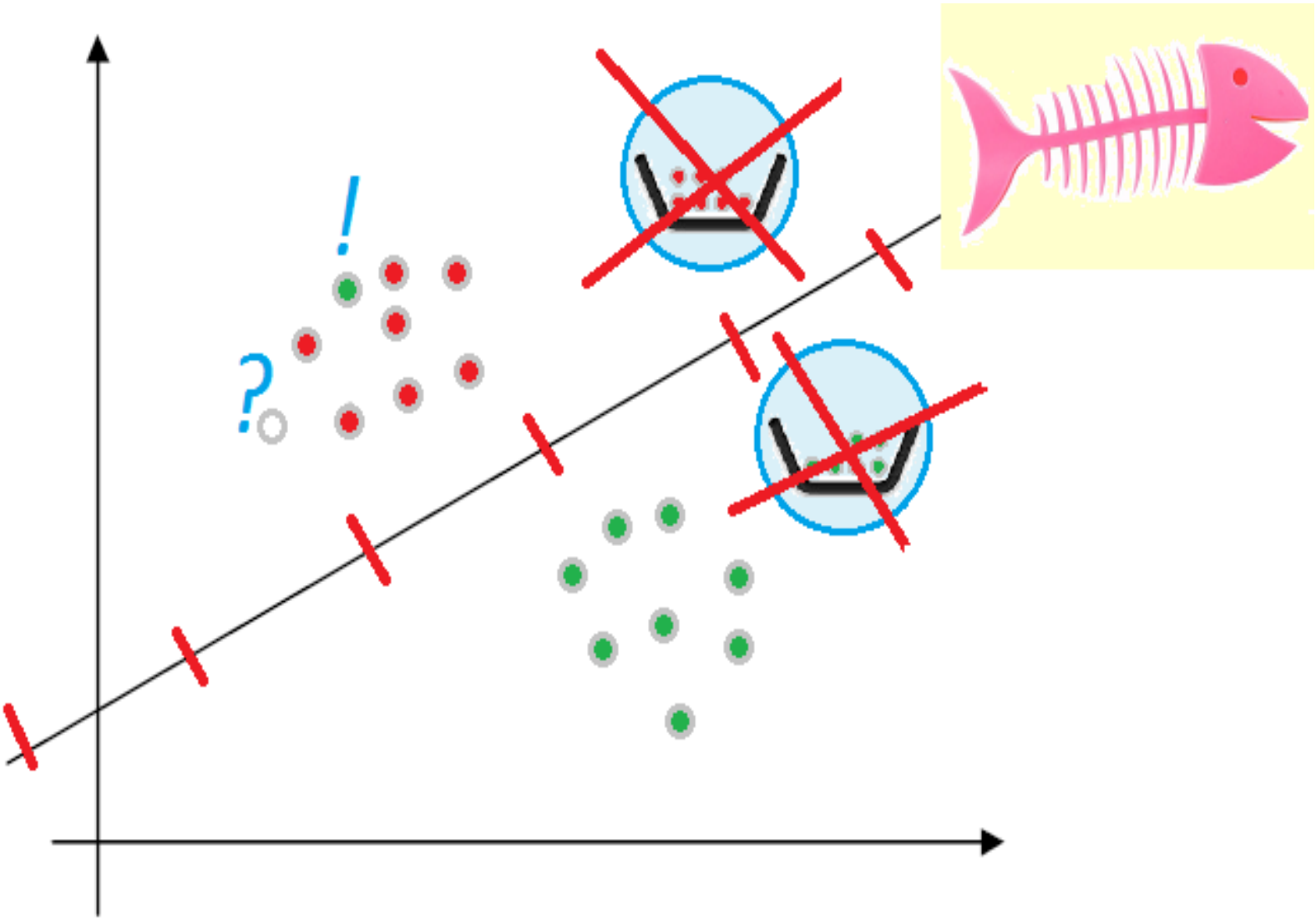


(0,1,1,0, 0,1,1,0, 1,1,0,0, 1,1,1,0)

Решающее правило



Решающее правило *контрпример*



Об эмпирических закономерностях

Законы

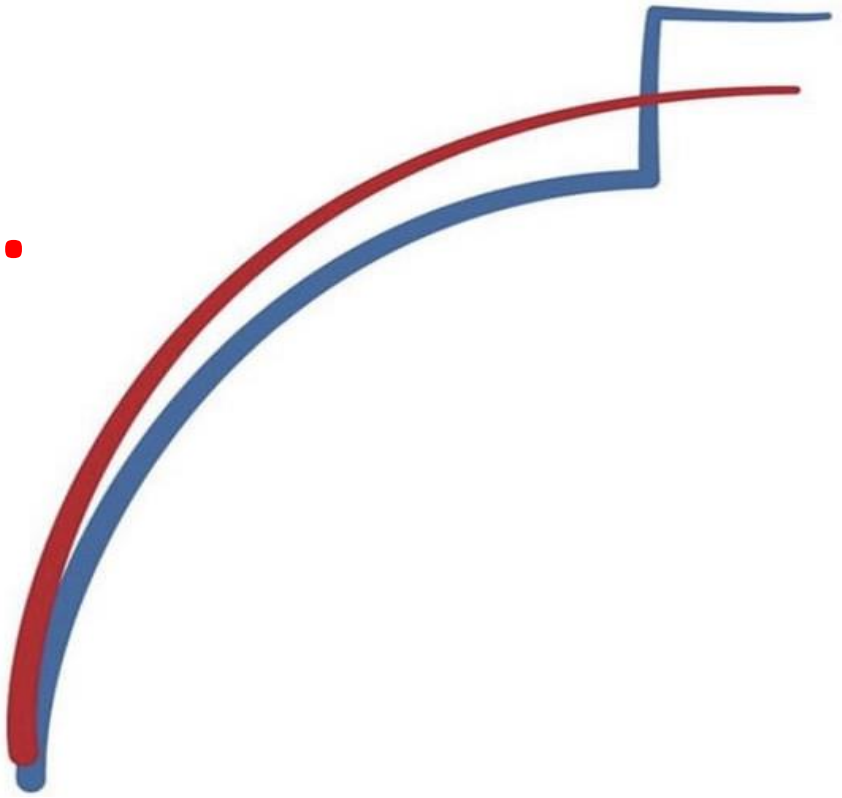
- Ципфа
- Брэдфорда

Принцип

- Парето (80/20)

и т.д.

VS.



Обозначения

Множество образов $\Omega = \{ \omega_1, \dots \}$

Индикаторы классов $M = \{1, \dots \}$

$\Omega = \Omega_1 \cup \dots \cup \Omega_m; \quad \Omega_i \cap \Omega_j = \varnothing$

Множество прецедентов $\Pi \subseteq M \times X$

Пространство признаков X // обычно $X = R^n$

в.о.с. $x(\omega) : \Omega \to X$

Решающее правило $g : X \to M, \quad g \in G$

Оценка качества $\Phi : G \to R$

Задачи распознавания образов

Задача классификации (= задача распознавания в узком смысле)

Распознавание с обучением ($П \neq \varnothing$) **vs.** Распознавание без обучения (кластеризация)

M конечно

Задача восстановления регрессии

M бесконечно

Задача прогнозирования

$\omega = \omega(t)$,
M – характеристики будущих моментов

Декомпозтция задачи распознавания

1. Подзадача генерации признаков

выбор признаков для описания образов

2. Подзадача селекции признаков

отбор наиболее информативных признаков

3. Подзадача построения решающего правила

4. Подзадача оценки системы <Признаки, Правило>

Общий подход к задаче обучения по прецедентам

по К.В.Воронцову

Дано	
X – пространство признаков ; ; ; ; M – индикаторы классов ; ; ; ; ; ; ; ; ; ; П – прецеденты	g – неизвестное решающее правило

Требуется	такой, что
алгоритм a(x)	$a(x) \approx g(x)$

Байесовская классификация

вероятностная постановка задачи классификации

Дано $M = \{1,2\}$; ; ; ; ; ; ; ; ; ; ; ; ; $P(\Omega_1), P(\Omega_2)$ – априорные вероятности принадлежности объекта классам ; ; ; ; ; ; ; $p(x\|\Omega_1), p(x\|\Omega_2)$ – функции распределения признаков для каждого класса ; ; ; ; ; ; Φ – минимум вероятности ошибки классификации	Известны*) вероятностные характеристики среды --- *) см. математическая статистика

Алгоритм**)

$g : X \to M$	такое, что $\Phi(g) \to \min$

**) максимум апостериорной вероятности

$$g(x) = 1 \Leftrightarrow p(x|\Omega_1)\, P(\Omega_1) > p(x|\Omega_2)\, P(\Omega_2)$$

Статистическая теория восстановления зависимостей

Вапник В.Н., Червоненкис А.Я. Теория распознавания образов (Статистические проблемы обучения). – М.: Наука, 1974.

Подход: X – *вероятностное пространство,*

$A = A(\theta_1 \dots \theta_s)$ – *семейство классификаторов,*

Найти лучший классификатор из A.
Что значит "лучший"?

$\upsilon(a)$ – *эмпирический риск, число ошибок* a *на прецедентах.*

$$? \, a^* \in A : \upsilon(a^*) = \min \{\upsilon(a) \mid a \in A\}$$

Главное: К-во прецедентов = ФУНКЦИЯ(h, ε, η),

$h = h(A)$ – *мера сложности семейства* $A \equiv$ *VC-размерность*);*

ε – *допустимое отклонение эмпирического риска* $\upsilon(a)$ *от истинного;*

η – *желаемое значение* $\upsilon(a)$

Статистическая теория восстановления зависимостей

Вапник В.Н., Червоненкис А.Я.
Теория распознавания образов
(Статистические проблемы обучения).
– М.: Наука, 1974.

	$\eta = 0.01$			
h	$\varepsilon = 0.01$	$\varepsilon = 0.05$	$\varepsilon = 0.1$	$\varepsilon = 0.2$
0	60106	2404	601	150
2	295074	9012	1946	408
5	673222	19884	4192	848
10	1307418	38160	7974	1589
20	2579359	74855	15572	3082
50	6401335	185193	38433	7575
100	12775769	369275	76581	15075

$\forall$ распределений вероятностей

$\forall$ классификаторов

Учет особенностей

Нежелательные явления в задачах распознавания с обучением

Переобучение ~~ ошибки классификации на тестовых примерах существенно выше ошибок на обучающих примерах.

Причины — Избыточно сложные модели

Недообучение ~~ существенная ошибка классификации на обучающих примерах.

Причины — Недостаточно сложные модели

Нормальное распределение признаков в классах

Инфо: *многомерная плотность нормального распределения*

$$N(x;\mu,\Sigma) = (2^n\pi^n|\Sigma|)^{-0.5}\exp(-0.5(x-\mu)^T\Sigma^{-1}(x-\mu))$$

$\mu \in R^n$ – *матем.ожидание*, $\Sigma \in R^{n\times n}$ – *ковар.матрица*

$$p(x|\Omega_i) = N(x;\mu_i,\Sigma_i)$$

ln | $p(x|\Omega_1)\,P(\Omega_1) > p(x|\Omega_2)\,P(\Omega_2)$

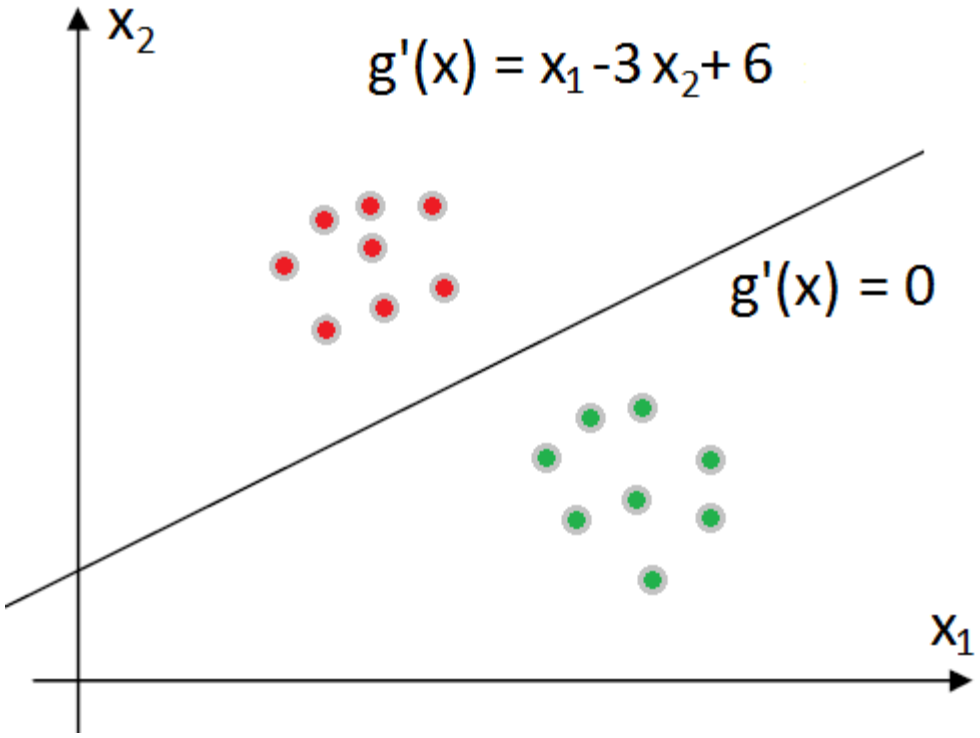
$\Rightarrow$ *Разделяющая гиперповерхность второго порядка*

- *Квадратичная разделяющая поверхность*
- *Линейная разделяющая поверхность*
  - *с диагональной* Σ | *с недиагональной* Σ | ...

Линейный классификатор

$g'(x) = w_1x_1 + \ldots + w_nx_n + w_0 = (w,x) + w_0$ такой, что

$g'(x) > 0$, если $x \in \Omega_1$ и $g'(x) < 0$, если $x \in \Omega_2$



Задача построения линейного классификатора

Дано $M = \{1,2\}$; ; ; ;

Π_1, Π_2 – конечные множества прецедентов

$\Pi_1 \subseteq \Omega_1, \quad \Pi_2 \subseteq \Omega_2$; ; ; ; ; ;

N.B. линейный классификатор существует

Алгоритм персептрона

вектор w

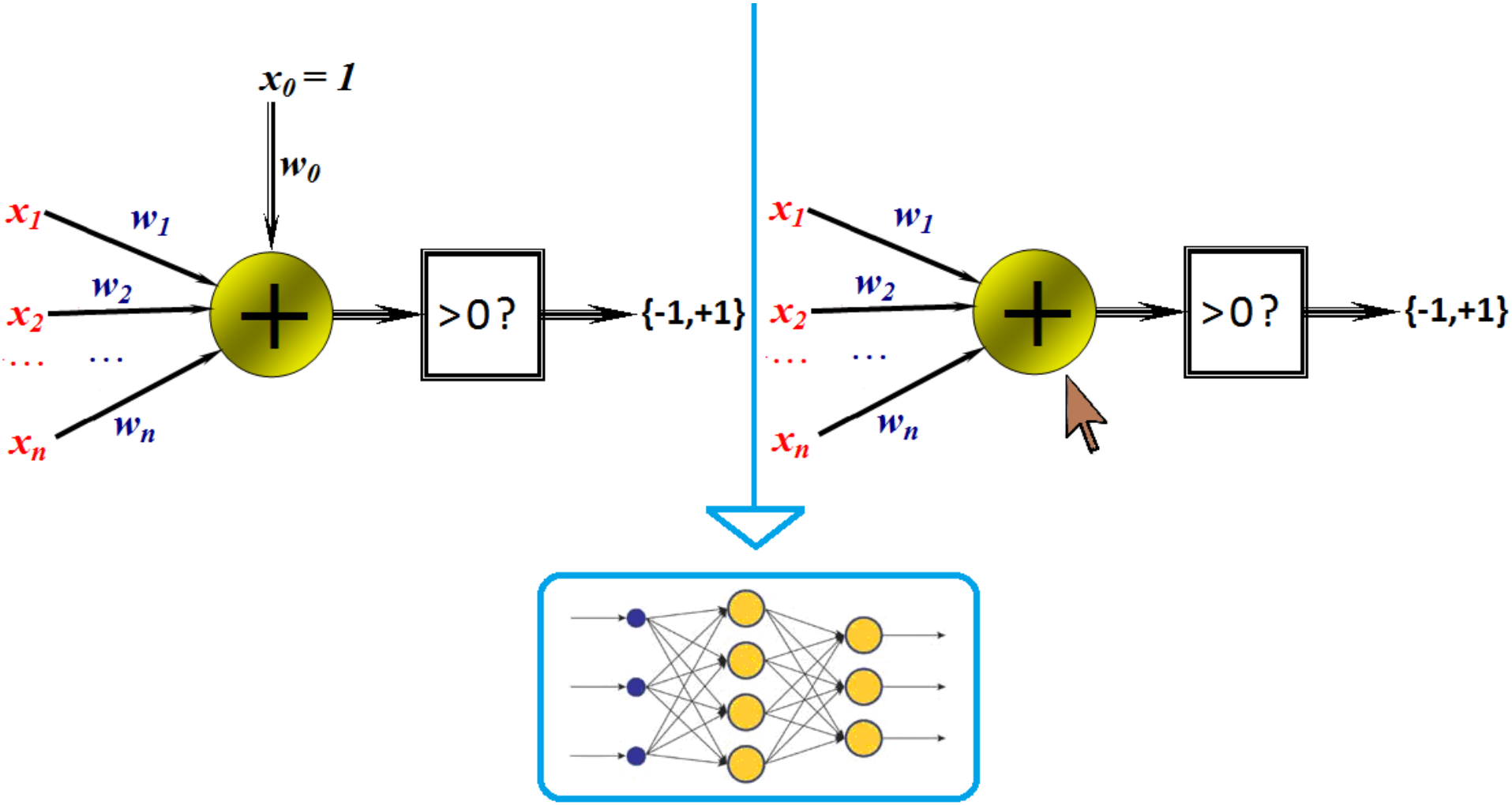
и число w_0

такие, что

$g'(x) > 0$, если $x \in \Pi_1$

$g'(x) < 0$, если $x \in \Pi_2$

Искусственный нейрон



Алгоритм персептрона

w = произвольное начальное значение

x_1 если $(w, x_1) < 0$, то $w = w + x_1$

x_2 если $(w, x_2) > 0$, то $w = w - x_2$

x_3 если $(w, x_3) > 0$, то $w = w - x_3$

x_4 если $(w, x_4) < 0$, то $w = w + x_4$

x_5 если $(w, x_5) < 0$, то $w = w + x_5$

...

Стоп, если w не меняется

Конечность алгоритма гарантирует теорема А.Новикова (1962, США):

Если w существует, то алгоритм заканчивает работу.

История линейных классификаторов

(+) Аркадьев А.Г., Браверман Э.М. Обучение машины распознаванию образов – М.: Наука, 1964

(–) Бонгард М.М. Проблема узнавания – М.: Наука, 1967

Банковский скоринг [Дюран, 1941]

как пример линейного классификатора

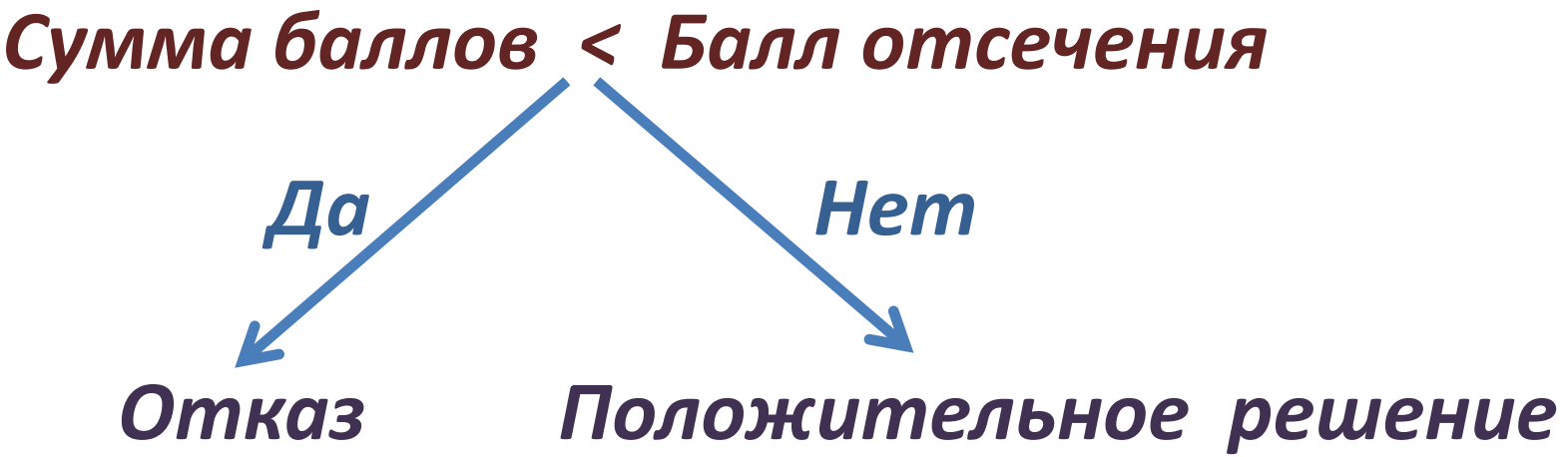
Скоринговая карта. Кредитование корпоративных клиентов.

		Scorecard Points
BRANCHE	BRANCHE = Missing, AGRICULTURE, AUTOMOTIVE, TELECOM, FINANCE, CHEMICAL	44
	BRANCHE = HIGH TECH	–20
	BRANCHE = OIL	53
COUNTRY	COUNTRY = Missing, NL, LU, FR, DE, AT	37
	COUNTRY = BE	52
DEBT	low ⇐ DEBT < 300000000	–23
	300000000 ⇐ DEBT < 500000000	59
	Missing, 500000000 ⇐ DEBT < high	121
LOYALITY	Missing, low ⇐ LOYALITY < 3	32
	3 ⇐ LOYALITY < 5	53
	5 ⇐ LOYALITY < 10	66
	10 ⇐ LOYALITY < high	80
PROFIT	Missing, low ⇐ PROFIT < 5000000	32
	7000000 ⇐ PROFIT < 10000000	37

Сумма баллов

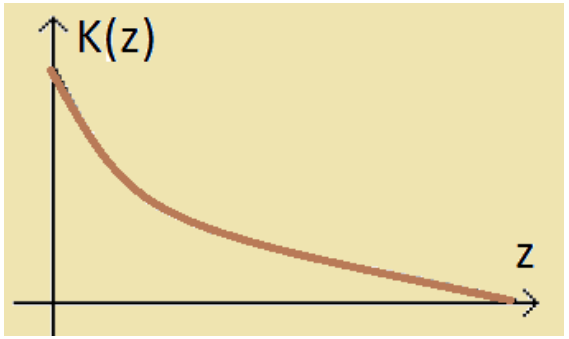
Банковский скоринг

принятие решения о кредитовании



Сумма кредита ~ Сумма баллов

Задача построения НЕлинейного классификатора

Дано $M = \{1,2\}$; ; ; ; Π_1, Π_2 – конечные множества прецедентов ; ; ; $K(z)$ – функция: $R \to R$; ; ; ; $d(x,y)$ – расстояние от x до y	$\Pi_1 \subseteq \Omega_1, \quad \Pi_2 \subseteq \Omega_2$; ; ; ; ; ; 

Метод потенциальных функций*)

Решающее правило g	

*) $$g(x) = 1 \Leftrightarrow \sum_{z \in \Pi_1} K(d(x,z)) > \sum_{z \in \Pi_2} K(d(x,z))$$

Айзерман М.А., Браверман Э.М., Розонэр Л.И. Метод потенциальных функций в теории обучения машин. – М.: Наука, 1970.

Подзадачи теории потенциальных функций

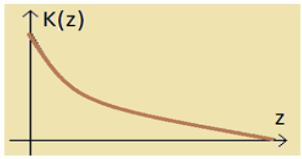
Дано $M = \{1,2\}$; ; ; ;

Π_1, Π_2 – конечные множества прецедентов ; ; ;

$K(z)$ – функция: $R \to R$; ; ; ;

$d(x,y)$ – расстояние от x до y

$\Pi_1 \subseteq \Omega_1, \quad \Pi_2 \subseteq \Omega_2$; ; ; ; ; ;



Метод потенциальных функций*)

Решающее правило g

*) $$g(x) = 1 \Leftrightarrow \sum_{z \in \Pi_1} K(d(x,z)) > \sum_{z \in \Pi_2} K(d(x,z))$$

?

? Выбор функций

? Сходимость

? Обучение

? Детерминистская постановка

? Вероятностная постановка

Нелинейный классификатор → *Линейный*

спрямляющее пространство

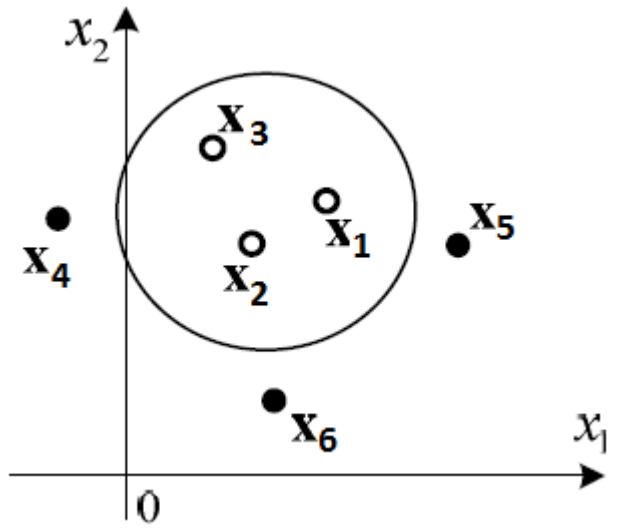
Дано $M = \{1,2\}$; ; ; ; Π_1, Π_2 – конечные множества прецедентов из R^n	Не существует линейный классификатор

? $R^n \rightarrow R^q$

Вектор чисел $w = (w_1 \dots w_q)$ Вектор-функция $f(x) = (f_1(x) \dots f_{q-1}(x), 1)$	$g(x) = (f(x),w)$ – классификатор для $f(\Pi_1)$ и $f(\Pi_2)$ в пространстве R^q

Существование **w** и **f(x)** доказано. Алгоритм?

Пример спрямляющего пространства



$$(x_1\text{-}1)^2 + (x_2 - 3)^2 - 2 = 0$$

$$x_1x_1+x_2x_2-2x_1-6x_2+8 =0$$

$R^2 \rightarrow R^6$

w = (+1, +1, 0, -2, -6, +8)

$x = (x_1, x_2)$

$f_1(x) = x_1 \times x_1$

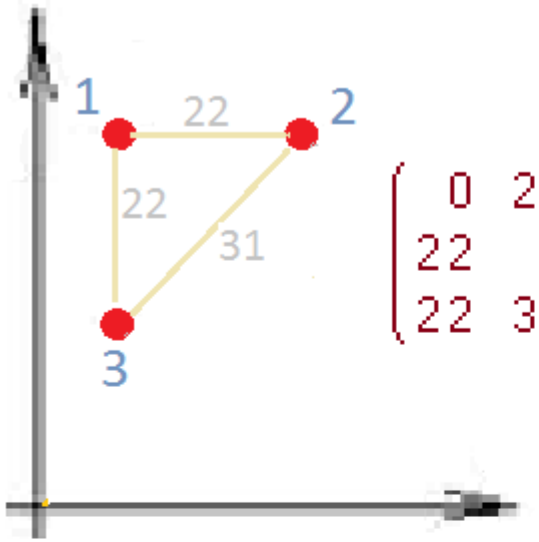
$f_2(x) = x_2 \times x_2$

$f_3(x) = x_1 \times x_2$

$f_4(x) = x_1$

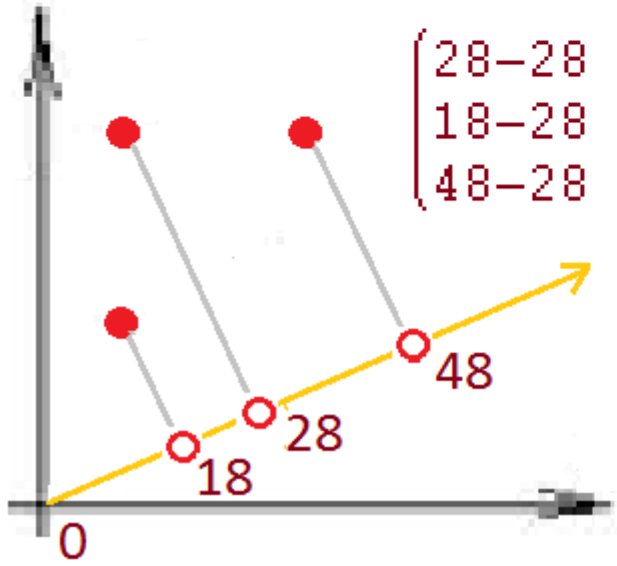
$f_5(x) = x_2$

Метод главных компонент Пролог



$$\begin{pmatrix} 0 & 22 & 22 \\ 22 & 0 & 31 \\ 22 & 31 & 0 \end{pmatrix}$$

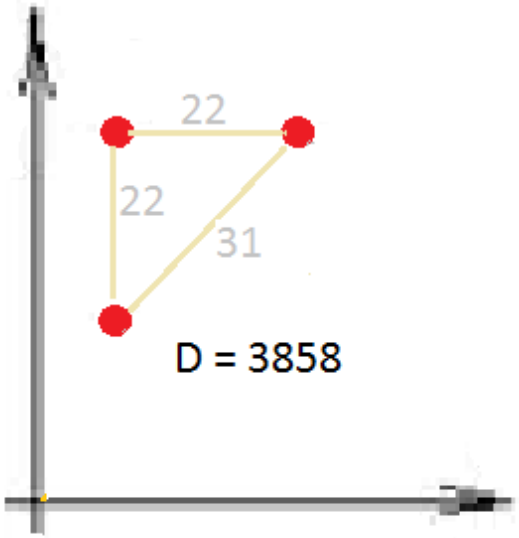
$$0^2+22^2+22^2+ 22^2+0^2+31^2+ 22^2+31^2+0^2=3858$$

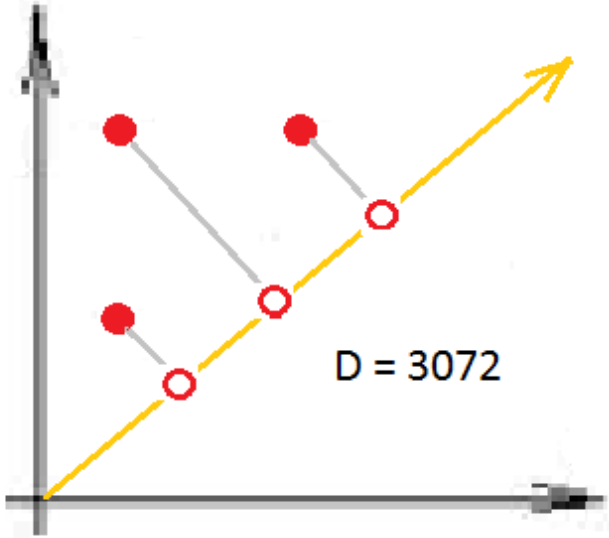


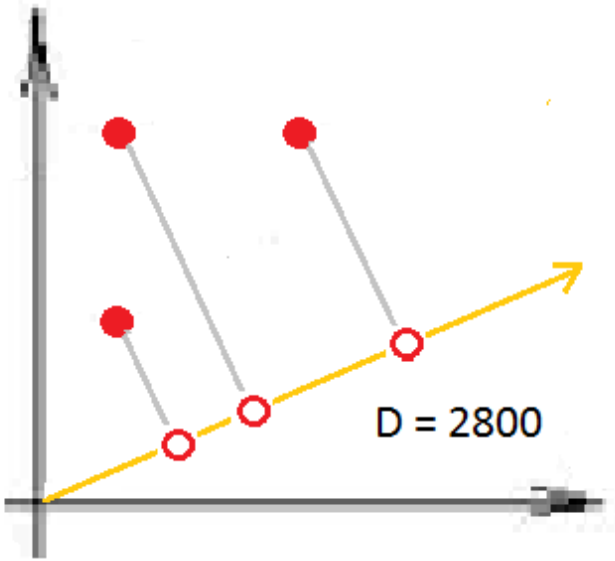
$$\begin{pmatrix} 28-28 & 28-18 & 28-48 \\ 18-28 & 18-18 & 18-48 \\ 48-28 & 48-18 & 48-48 \end{pmatrix} = \begin{pmatrix} 0 & 10 & -20 \\ -10 & 0 & -30 \\ 20 & 30 & 0 \end{pmatrix}$$

$$0^2+10^2+20^2+ 10^2+0^2+30^2+ 20^2+30^2+0^2=2800$$

Метод главных компонент [Пирсон, 1905]



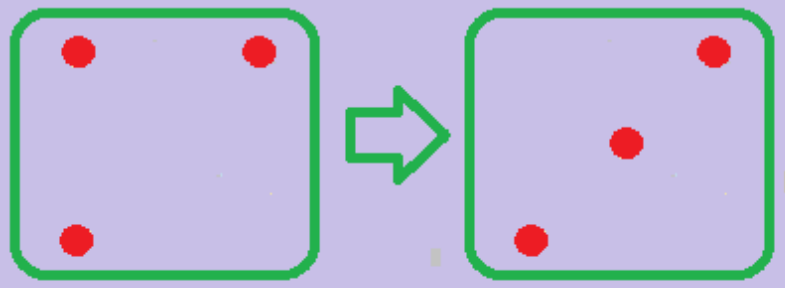




$$D = \sum_{i=1}^{N} \sum_{j=1}^{N} d^2(i,j)$$

$$D(e) \to \max \quad \equiv \quad Ce = \lambda_{max}\, e,$$

где $C = C(x_1 \dots x_N)$ – ков/кор матрица



сохр = 3072/3858 = 0.8

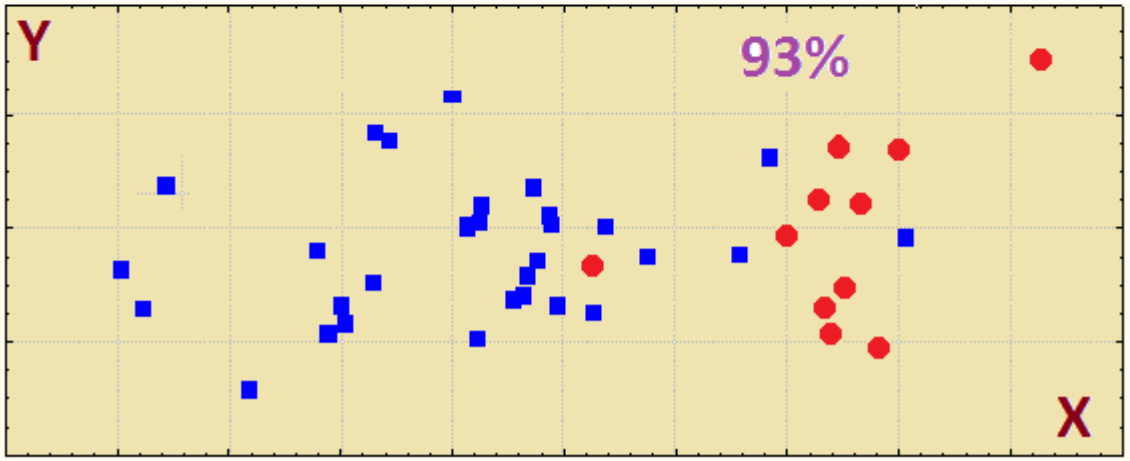
потери = 786/3858 = 0.2

?

Метод главных компонент (Иллюстрация)

	Рост	Вес	Возраст	Доход	IQ	
1						
2						
n						

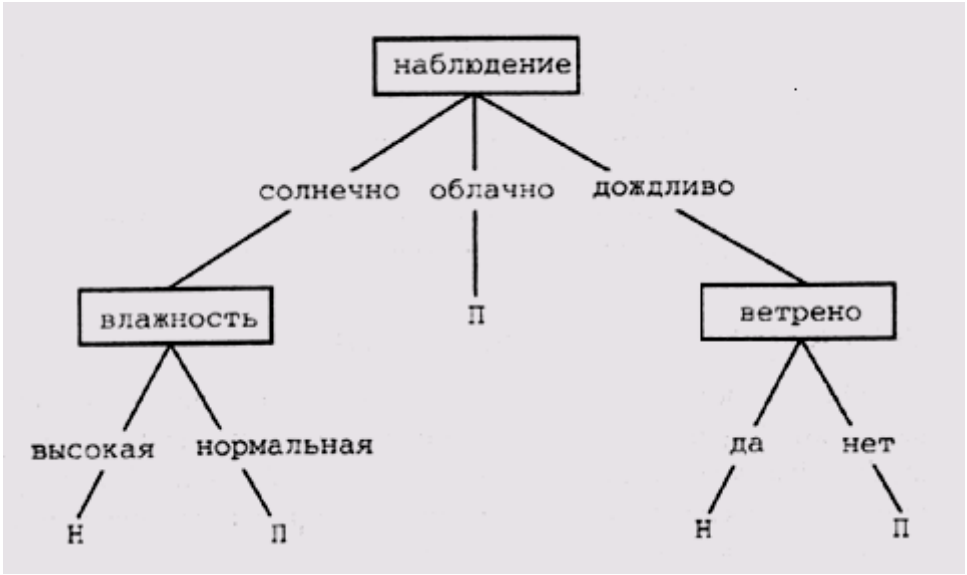
X	-0.08	-0.32	0.79	-0.39	-0.34	D= 989	84%
Y	0.41	0.12	0.08	-0.60	0.67	D= 121	9%



-0.08Рост-0.32Вес+0.79Возраст-0.39Доход-0.34IQ

Решающее дерево как классификатор

Решающее дерево – иерархически организованная система вопросов и сопоставленных им ответов, позволяющая получать классификационные решения.



Задача построения решающего дерева

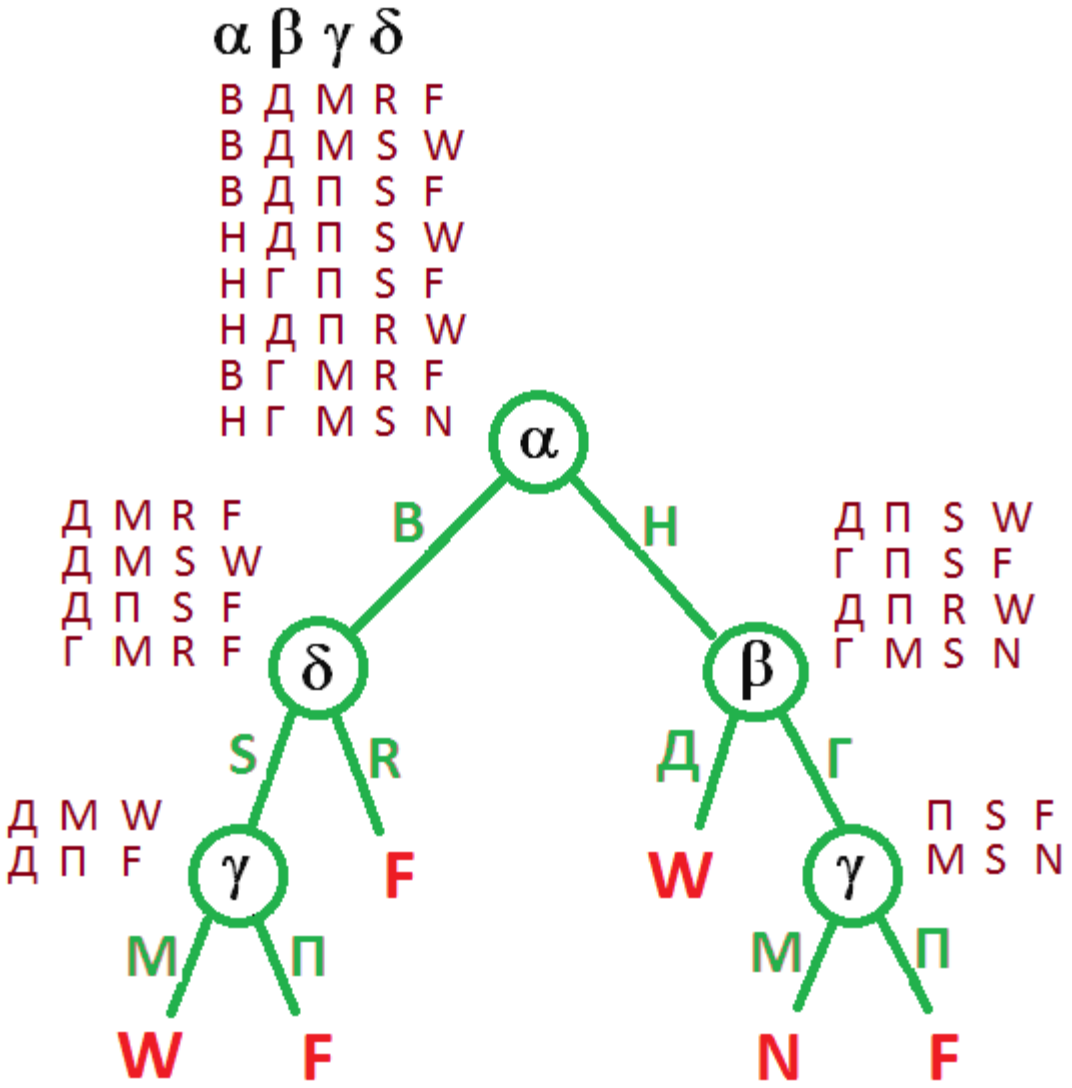
Дано П – прецеденты ;;;;;;;	
Критерий максимально информативного ветвления дерева	

ID3 | C4.4 | CART

Требуются	такие, что
Решающее дерево	

Задача построения решающего дерева

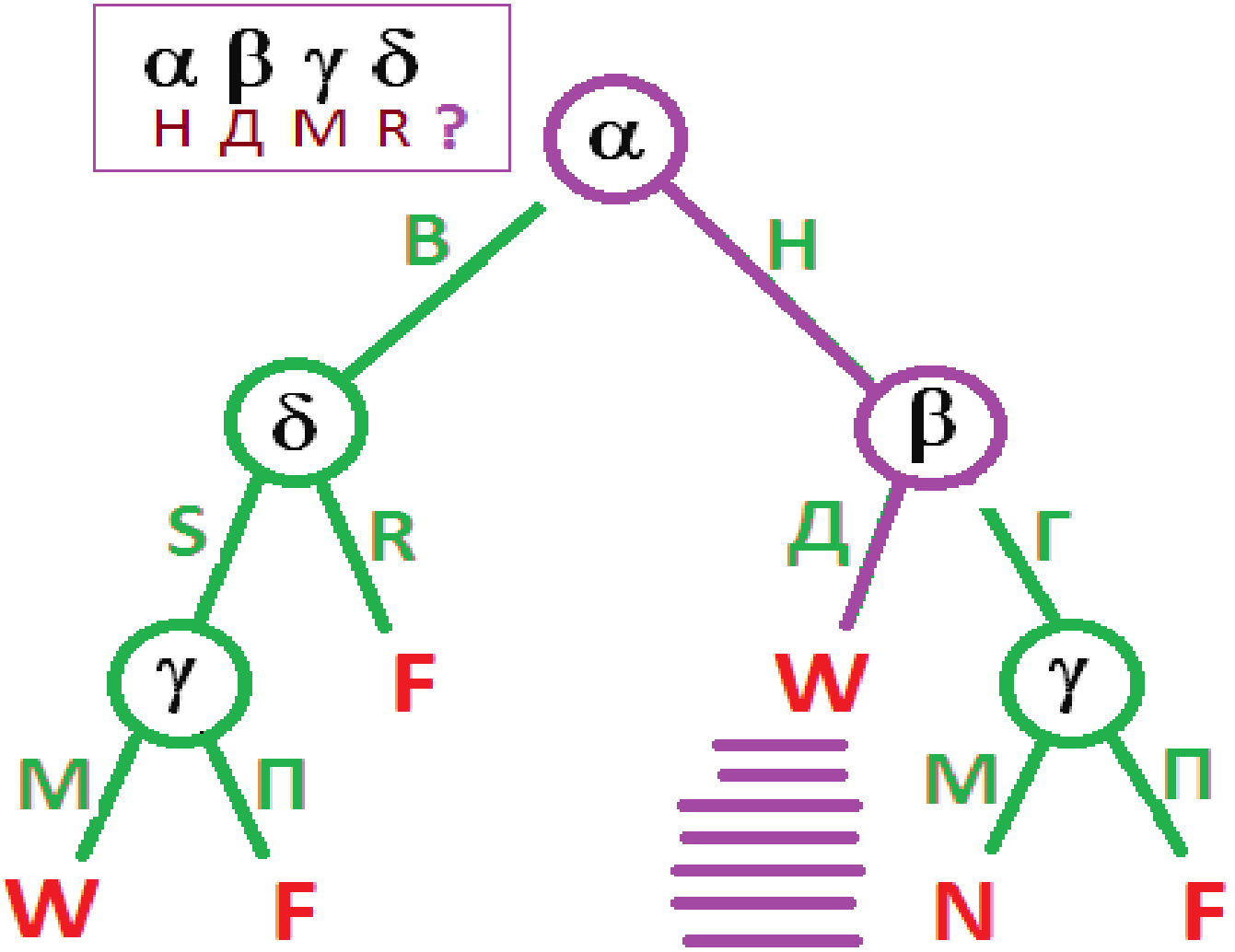
Пример



Редукция – удаление поддеревьев, имеющих недостаточную статистическую надежность.

Применение решающего дерева

Пример



Задача композиции классификаторов

Дано X – пространство признаков ; ;

$M=\{-1,+1\}$ – индикаторы классов ; ;

$П$ – прецеденты ; ; ; ; ; ; ; ; ; ; ; ;

$a_1, \ldots, a_T$ – базовые классификаторы;

$Ф$ – функционал качества композиции

композиция:

$\text{sign}(\alpha_1 a_1(x) + \ldots + \alpha_T a_T(x))$;

$Ф =$ число ошибок на прецедентах ; ; ;

N.B. *)

бустинг | бэггинг | голосование | смеси

Требуются

коэффициенты $\alpha_1, \ldots, \alpha_T$

композиции $a(x)$

такие, что

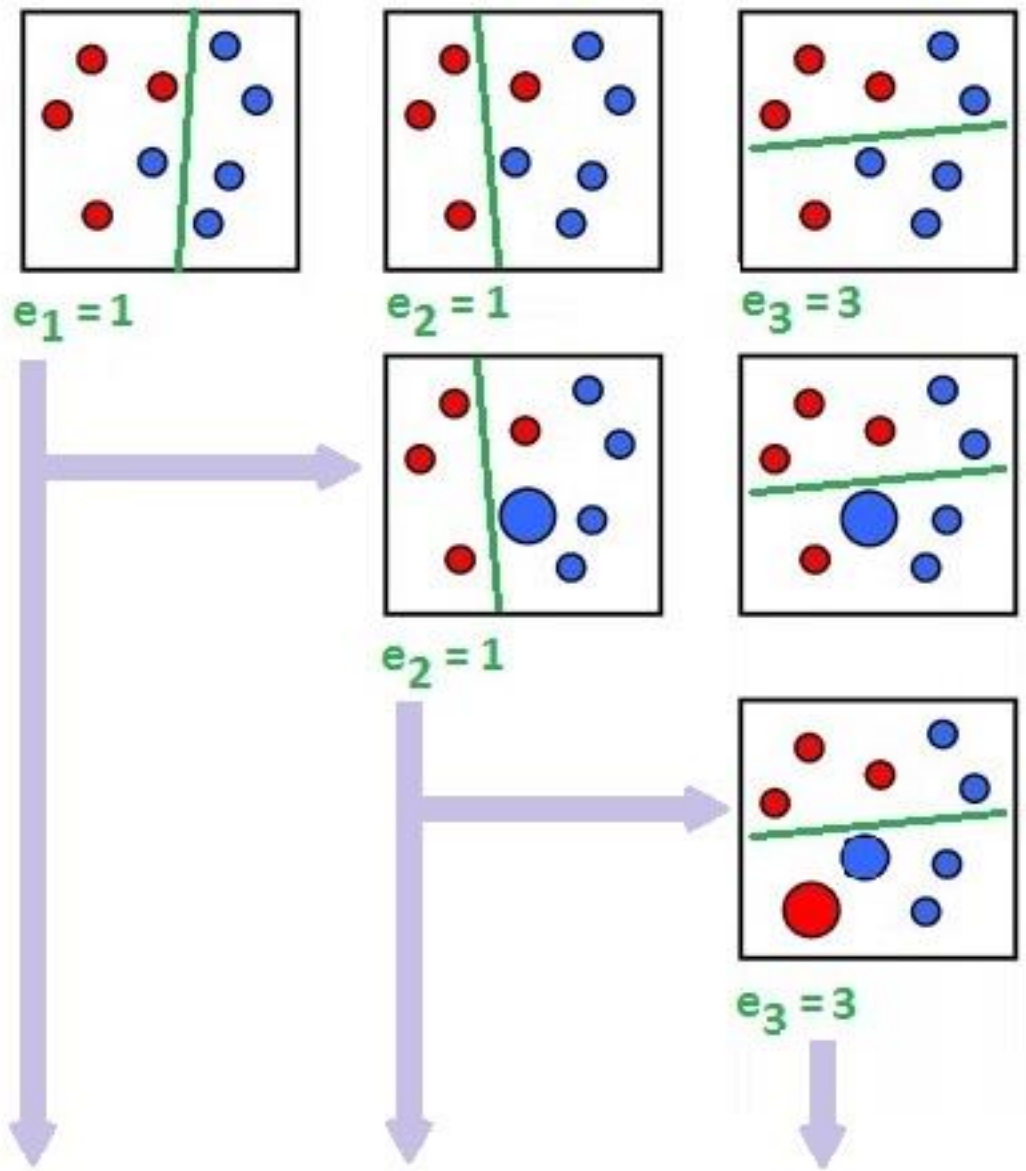
$$Ф(a(x)) \to \min$$

*) $a_1, \ldots, a_T$ классифицируют прецеденты лучше, чем наугад для любого распределения весов прецедентов.

Алгоритм AdaBoost [Schapire,1996]

Размер значка ~
Вес прецедента

T = 3

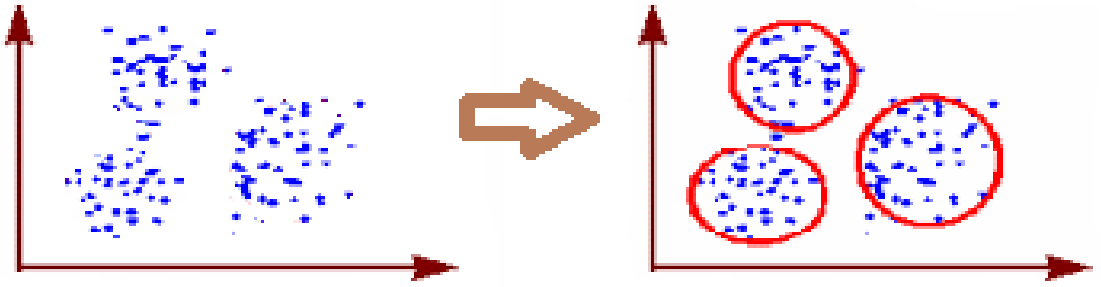


$$sign(\alpha_1 a_1(x) + \alpha_2 a_2(x) + \alpha_3 a_3(x))$$

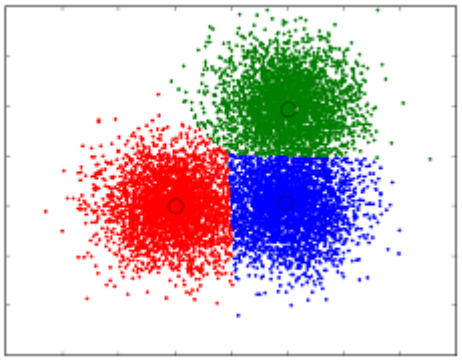
Общий подход к задаче кластеризации (к задача распознавания без обучения)

по Л.М.Местецкому

Дано $x_1, \ldots, x_n$ – множество векторов-признаков для набора признаков неизвестной классификации	
Требуется разделить $x_1, \ldots, x_n$ на классы по сходству векторов-признаков.	



VS.



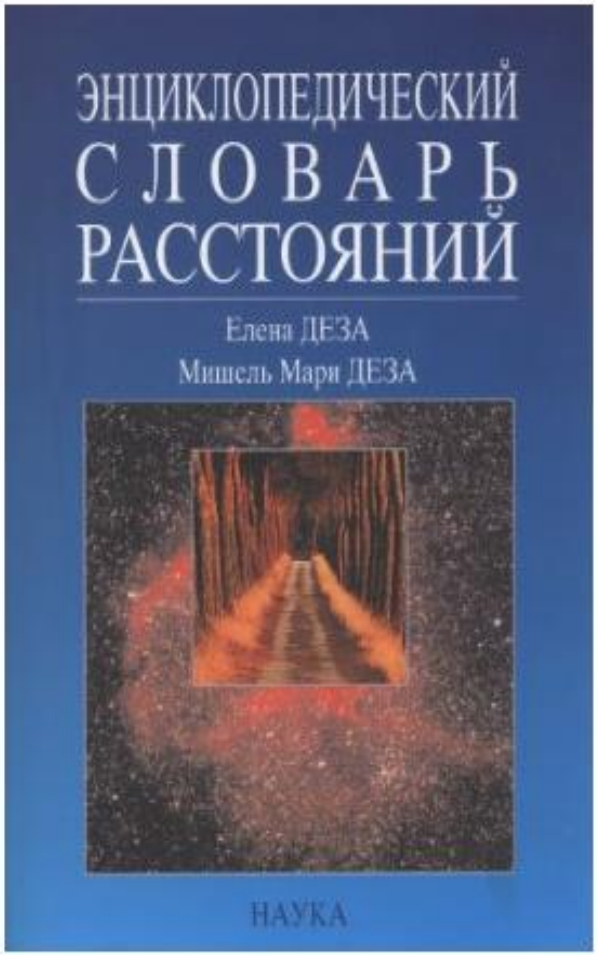
Меры сходства

Евклидова метрика: $$d(a,b) = \sqrt{\sum_{k=1}^{n} (a_k - b_k)^2}$$

Манхэттенское расстояние: $$d(a,b) = \sum_{k=1}^{n} |a_k - b_k|$$

Коэффициент Жаккара: $$d(A,B) = \frac{|A \cap B|}{|A \cup B|}$$

Коэффициент Дайса: $$d(A,B) = \frac{2|A \cap B|}{|A| + |B|}$$



Базовые алгоритмы кластеризации

Иерархические

- Агломеративные (слияние близких)
- Дивизимные (разбиение)

Неиерархические

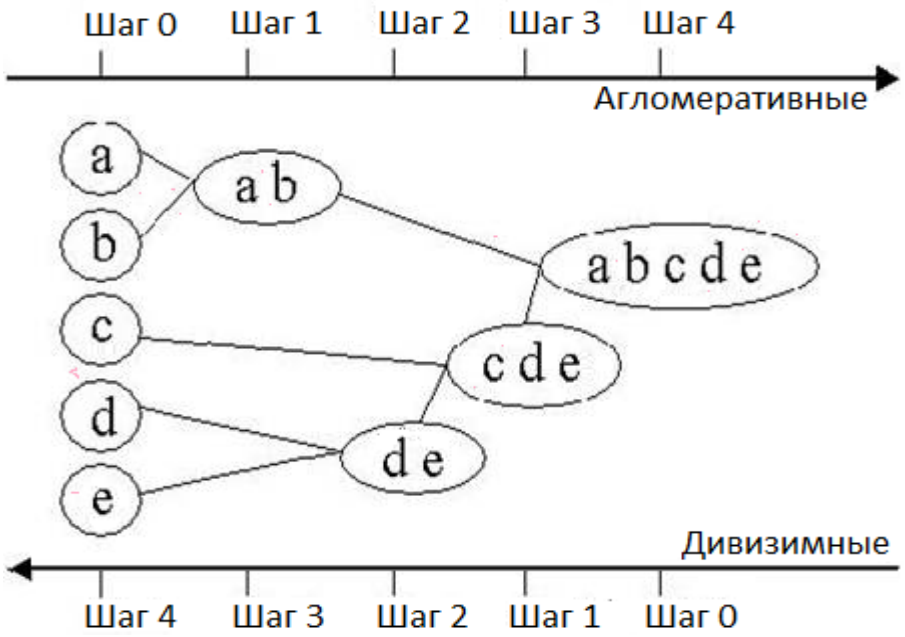
- Графовые
- Статистические
- Алгоритмы квадратической ошибки
  - FOREL | k-means | …

Задача иерархической кластеризации

Дано $x_1, \ldots, x_n$; ; ; ; ;
d(x,y)

Алгоритмы иерархической кластеризации

Требуется
Иерархическая структура



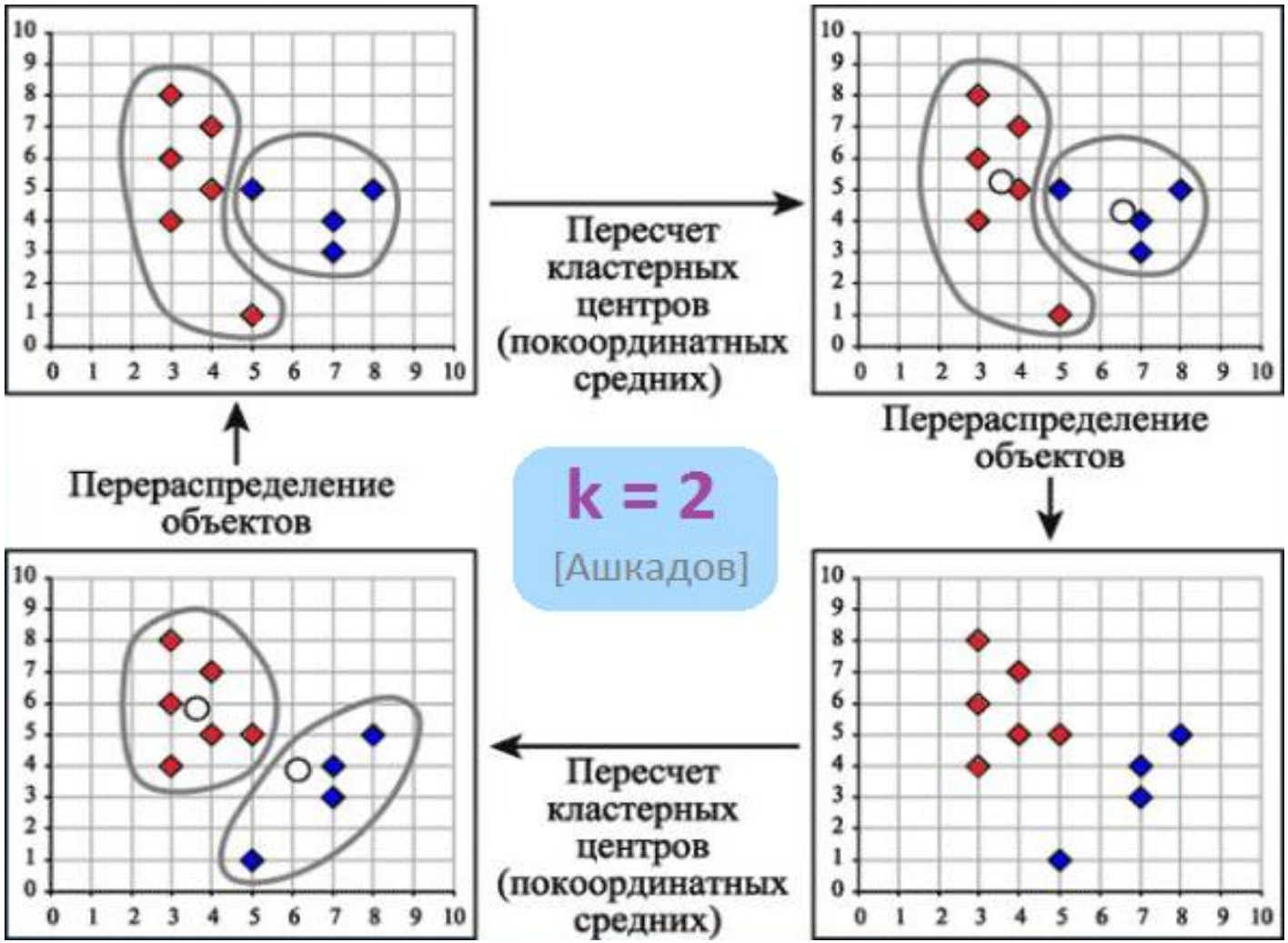
Задача кластеризации методом k-means

Дано $x_1, \ldots, x_N$; ; ; ; ; ; $d(x,y)$; ; ; ; ; ; ; число k	X^n – линейное пр-во ; ; ; ; ; $d(x,y)$ – метрика

Алгоритм k-means

Требуется k кластеров	

Задача кластеризации методом k-means



Кластеризация на графах

Репозиторий UCI

archive.ics.uci.edu/ml/datasets.html

Более 300 обучающих выборок

Определение местоположения протеинов в клетке.
Объектов: 336, признаков: 7, классов: 8

poligon.machinelearning.ru

В о п р о с ы?

soloviev@glossary.ru

Соловьев С.Ю. Постановки задач современной информатики.
www.park.glossary.ru