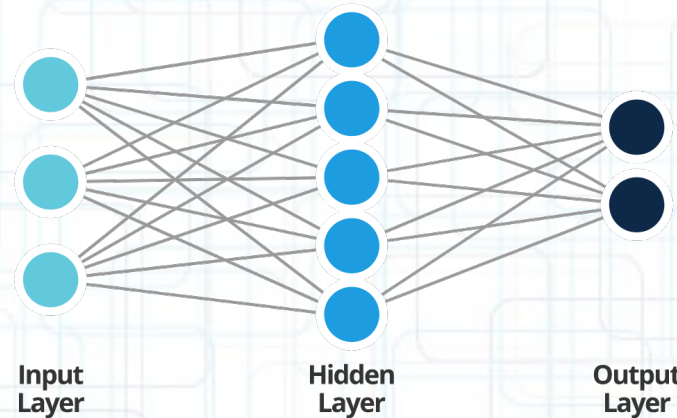
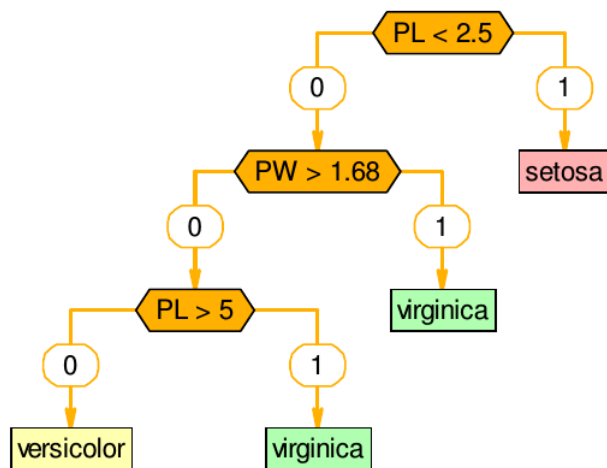
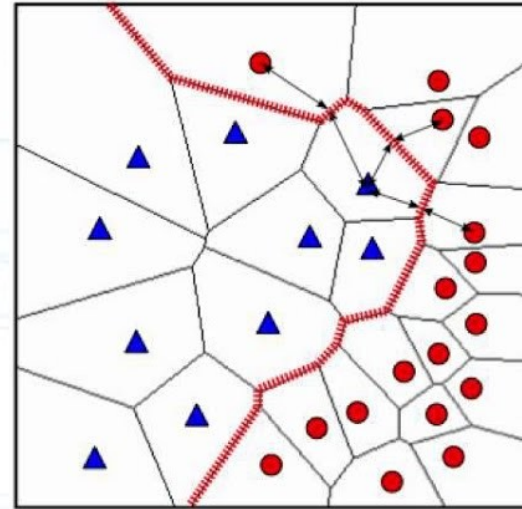
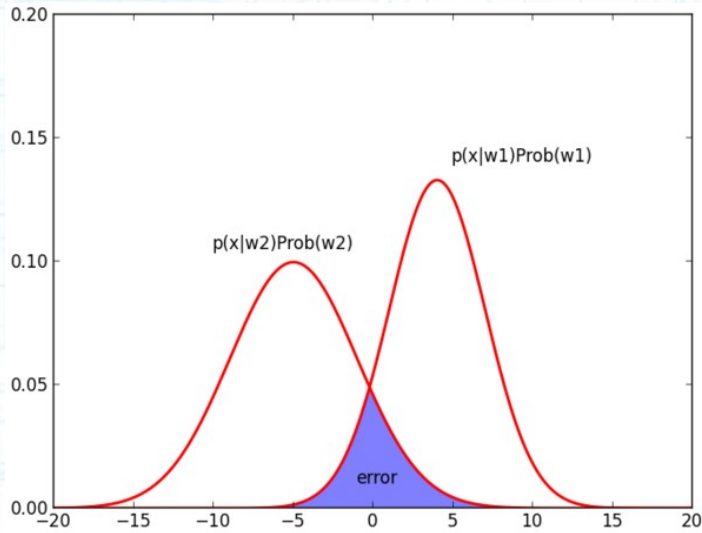
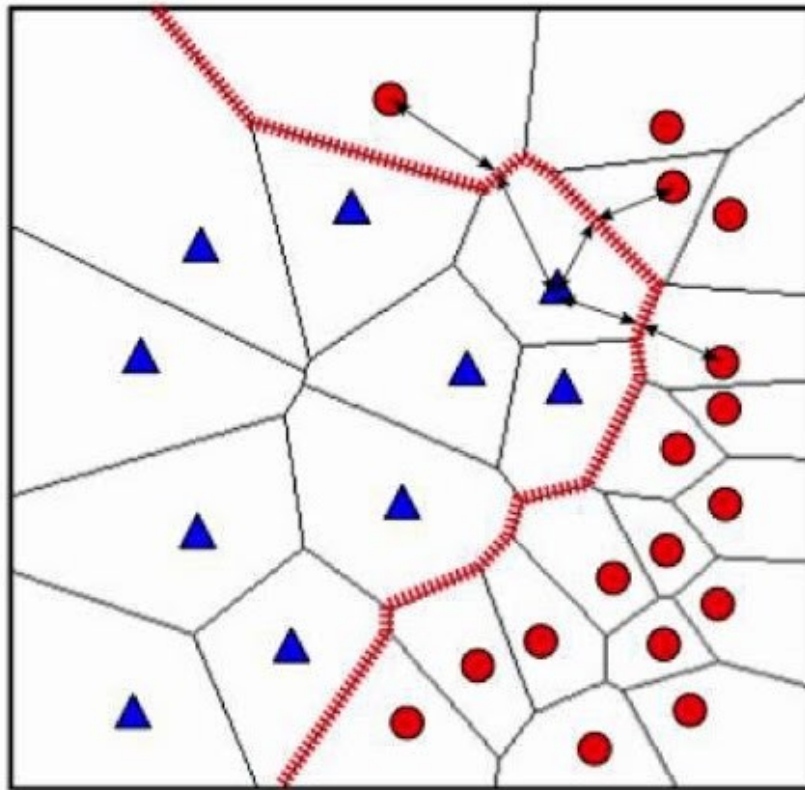


Алгоритмы машинного обучения



Метрические алгоритмы



Гипотезы

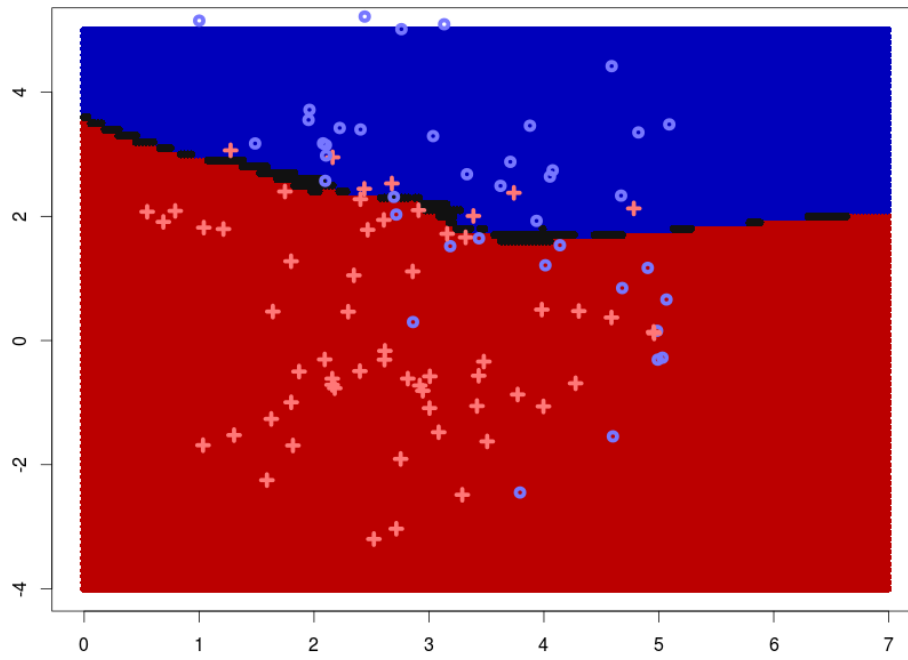
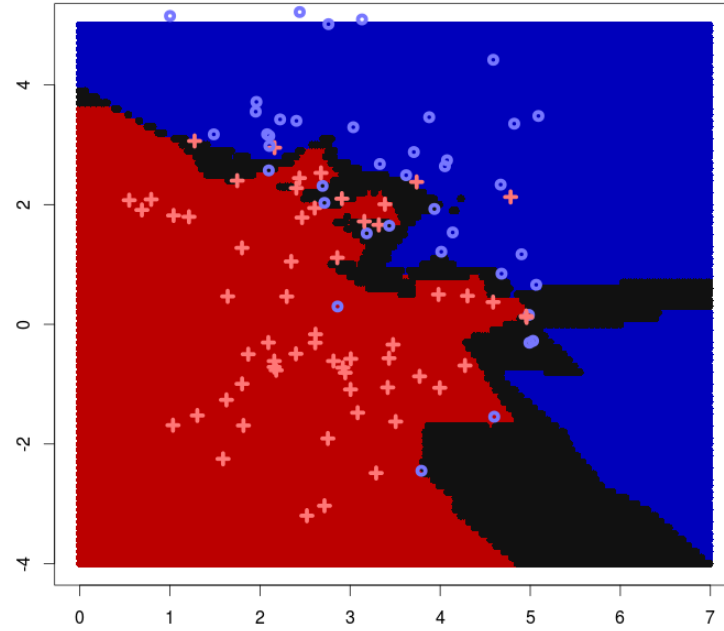
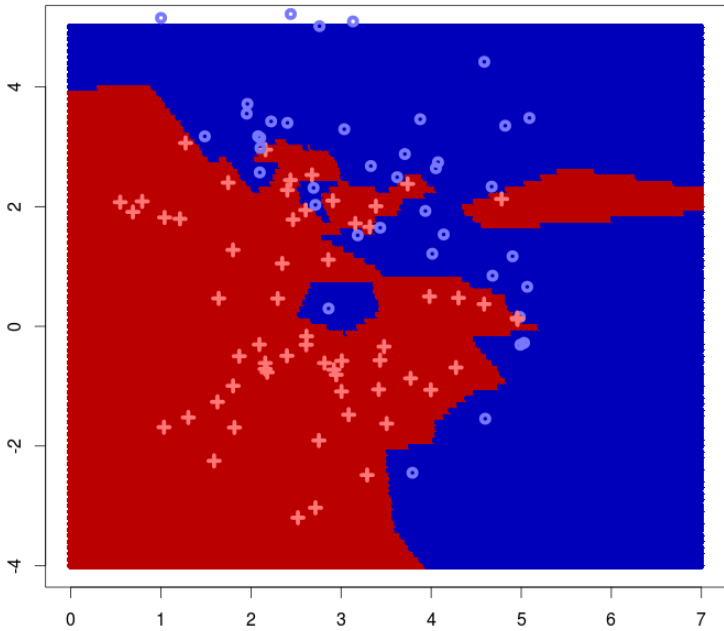
- Задачи классификации и регрессии:
 - X — объекты, Y — ответы;
 - $X_\ell = (x_i, y_i)$ — обучающая выборка;
- Гипотеза компактности (для классификации):
 - Близкие объекты, лежат в одном классе.
- Гипотеза непрерывности (для регрессии):
 - Близким объектам соответствуют близкие ответы.
- Формализация понятия «близости»:
 - Задана функция расстояния $\rho : X \times X \rightarrow [0, \infty)$.
- Пример. Евклидово расстояние и его обобщение:

$$\rho(x, x_i) = \left(\sum_{k=1}^n |x^{(k)} - x_i^{(k)}|^2 \right)^{\frac{1}{2}} \quad \rho(x, x_i) = \left(\sum_{k=1}^n \beta_k |x^{(k)} - x_i^{(k)}|^p \right)^{\frac{1}{p}}$$

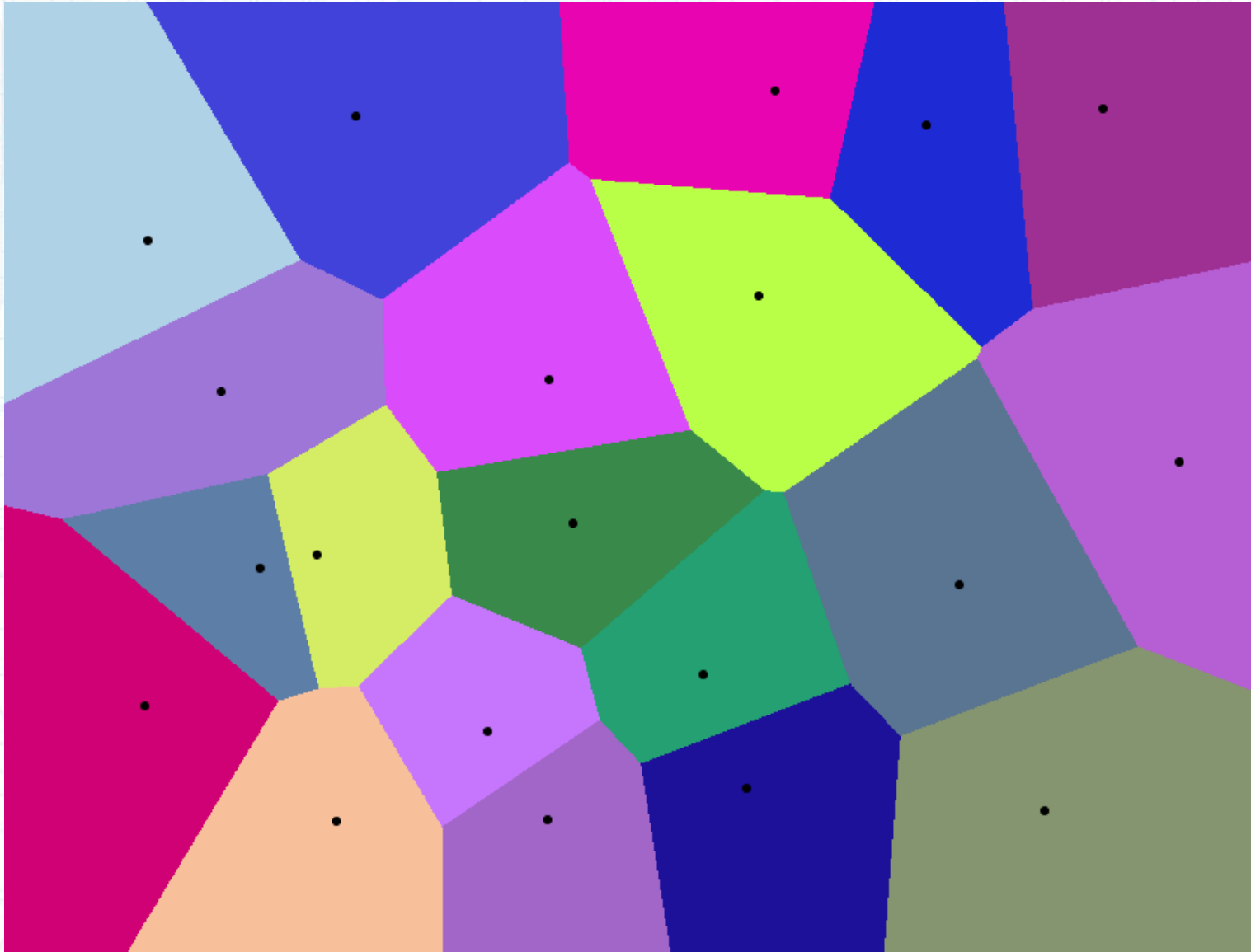
Lazy learning

- Это так называемое ленивое обучение, в котором нет этапа тренировки параметров модели. Сразу происходит этап предсказания.
- Подходит для задач, в которых сложно сформулировать набор признаков, но легко сравнивать объекты (пример: сравнительная геномика)
- Недостаток: медленный процесс предсказания

Методы 1-го, 4-х, 60-и ближайших соседей



Диаграммы Вороного



Метод окна Парзена

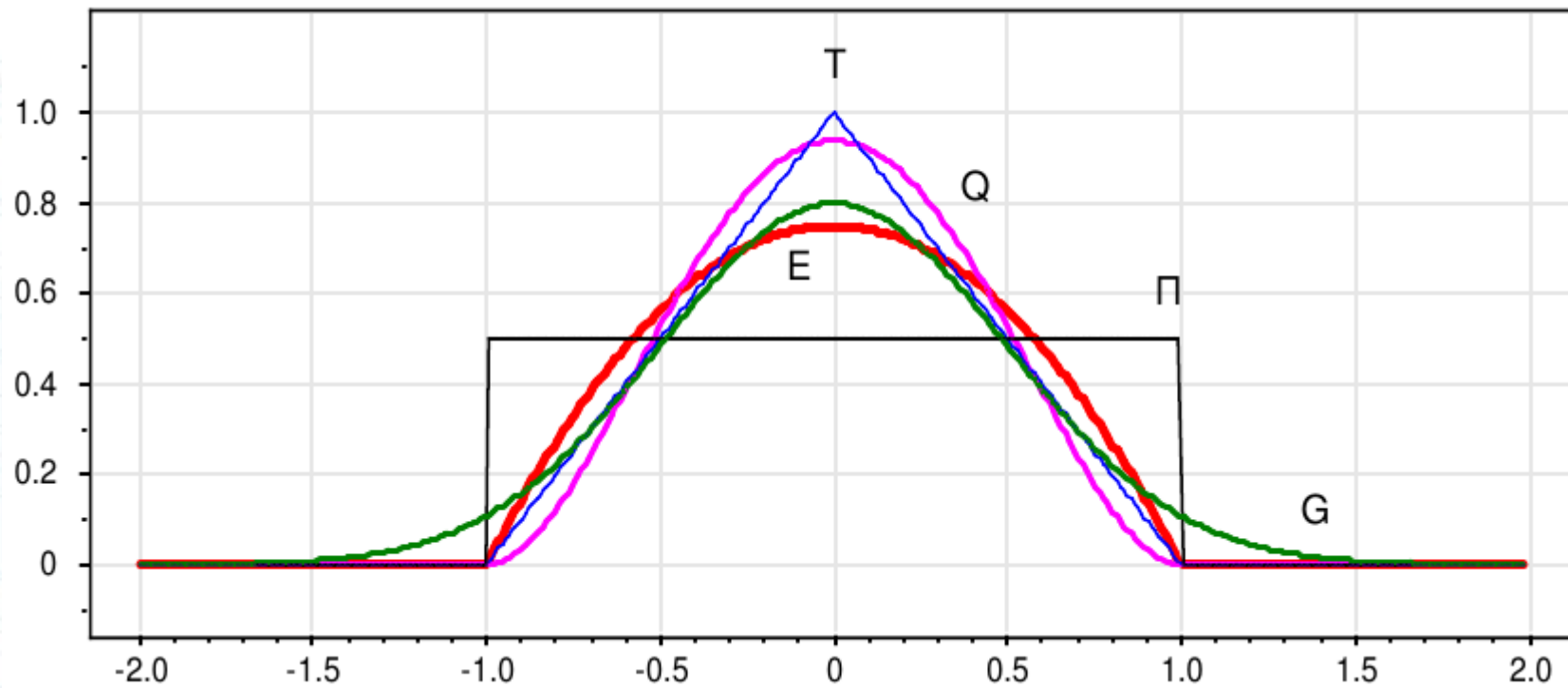
Вес соседей w задается с помощью неотрицательной невозрастающей функции K от расстояния до соседа. Сумма весов соседей класса трактуется как вероятность этого класса.

$$w(i, x) = K\left(\frac{\rho(x, x^{(i)})}{h}\right), \text{ где } h \text{ — ширина окна,}$$
$$K(r) \text{ — ядро, не возрастает и положительно на } [0, 1]$$

При фиксированной ширине окна качество классификатора сильно зависит от плотности точек.

Выход: положить ширину h равной расстоянию до k -того соседа

Часто используемые ядра



$\Pi(r) = [|r| \leq 1]$ — прямоугольное

$T(r) = (1 - |r|) [|r| \leq 1]$ — треугольное

$E(r) = (1 - r^2) [|r| \leq 1]$ — квадратичное (Епанечникова)

$Q(r) = (1 - r^2)^2 [|r| \leq 1]$ — четвертое

$G(r) = \exp(-2r^2)$ — гауссовское

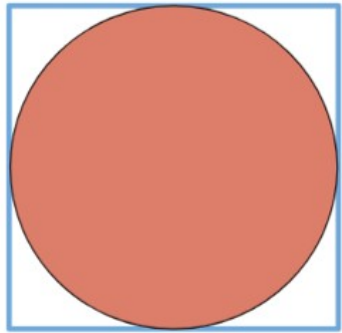
Проклятие размерности

- Проклятие размерности - усреднение значений метрики при большом количестве признаков. Почти до всех ближайших соседей расстояние одинаково
- Почему это происходит:
 - Шар радиуса R имеет объем $V(R) \sim R^D$
 - Объем шара радиуса 0.9 в 20-мерном пространстве составляет всего 12% от объема шара радиуса 1.
Т.е. 88% точек лежит на сфере: $0.9 < R < 1$

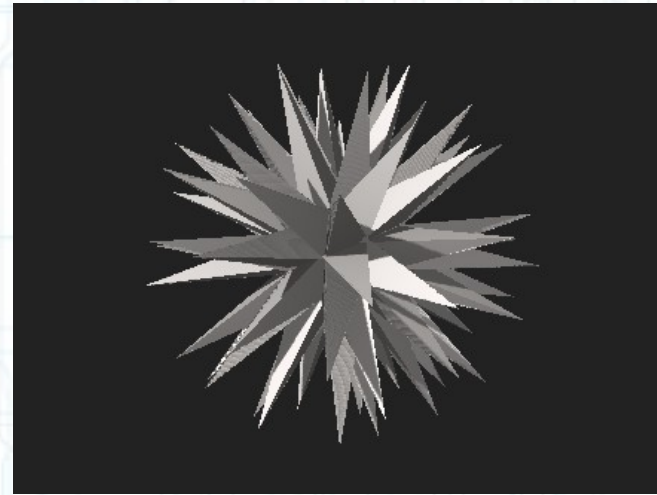
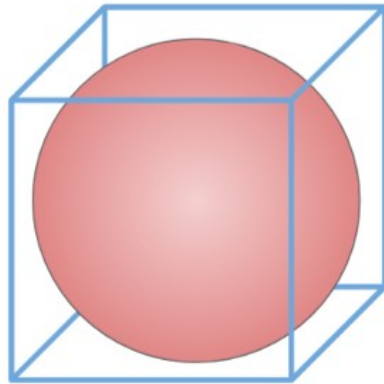
$$\frac{V(R - \varepsilon)}{V(R)} = \left(\frac{R - \varepsilon}{R} \right)^D \xrightarrow{D \rightarrow \infty} 0$$

Проклятие размерности

A

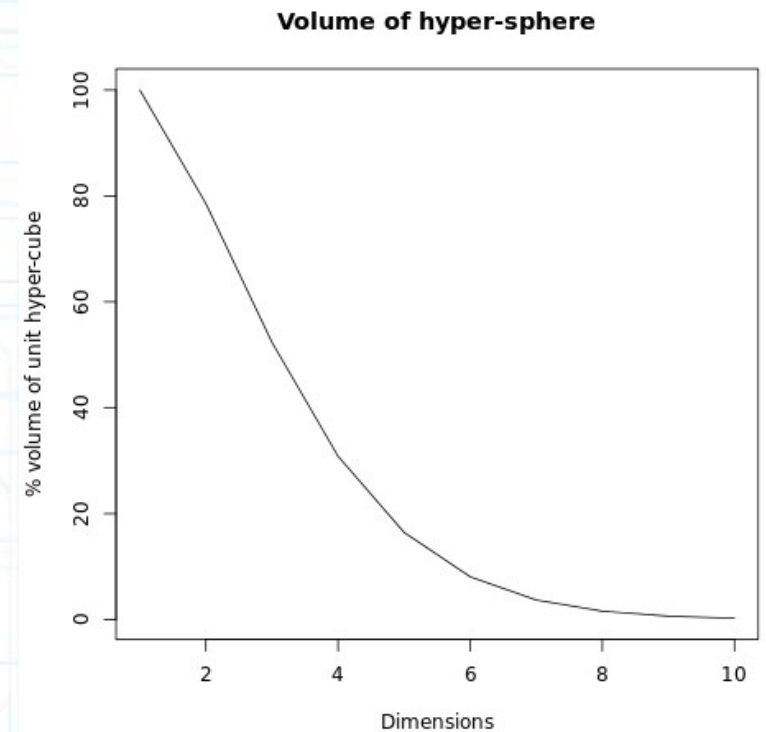


B



Объем вписанной в куб сферы в многомерном пространстве во много раз меньше объема куба!

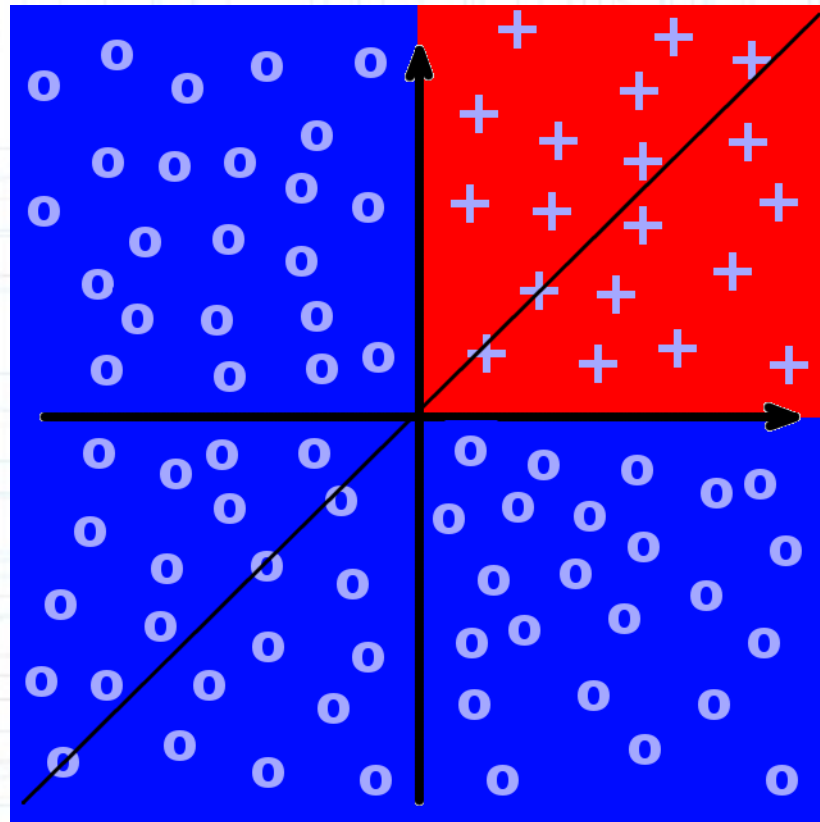
Расстояние до вершины куба: \sqrt{n} . Количество вершин: 2^n



Проклятие размерности

Пример

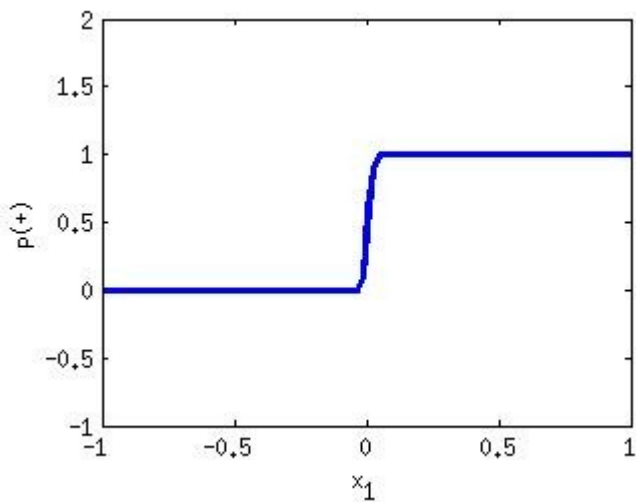
- Пространство признаков: \mathbb{R}^n .
- Класс +: область $x_{1,2} > 0$ (остальные координаты произвольны)
- X_ℓ - равномерно распределена



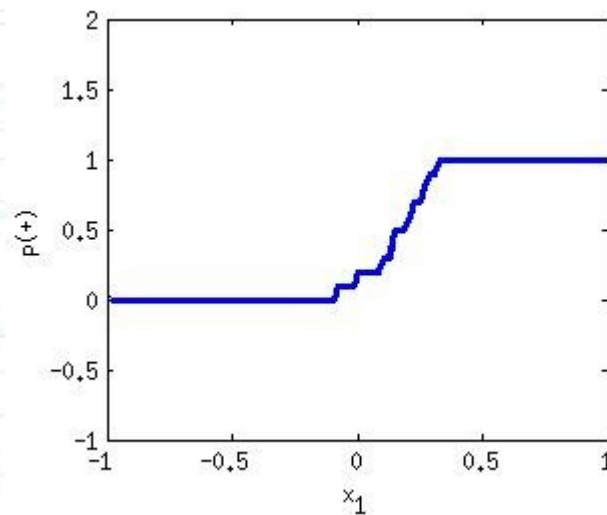
Проклятие размерности

Пример

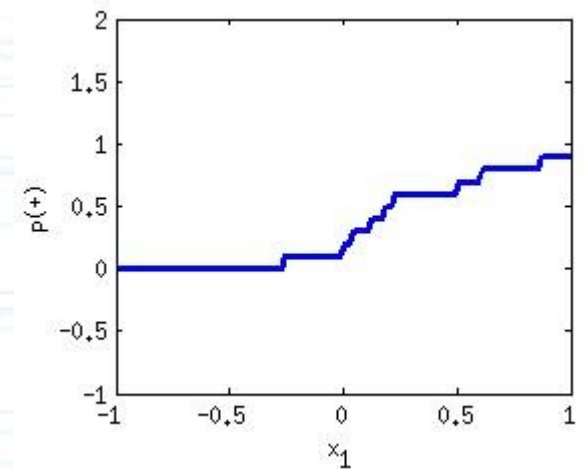
- Метод 10 ближайших соседей. $\ell = 10000$
- Относительная частота класса “+” на прямой: $x_1 = x_2$, $x_3 = 0$, $x_4 = 0, \dots$



$n=2$



$n=5$



$n=20$

Вывод: для больших размерностей метрические алгоритмы сглаживают границы областей классов

Жадное добавление признаков

1) Вдруг одного признака достаточно?

Расстояние по k-му признаку: $\rho(x, x_i) = |x^{(k)} - x_i^{(k)}|$

Выберем наилучший признак: $LOO(k) \rightarrow \min_k$

2) Добавим еще один признак k:

$$\rho^p(x, x_i) := \rho^p(x, x_i) + \beta_k |x^{(k)} - x_i^{(k)}|^p$$

Найдем лучший k и коэффициент β_k

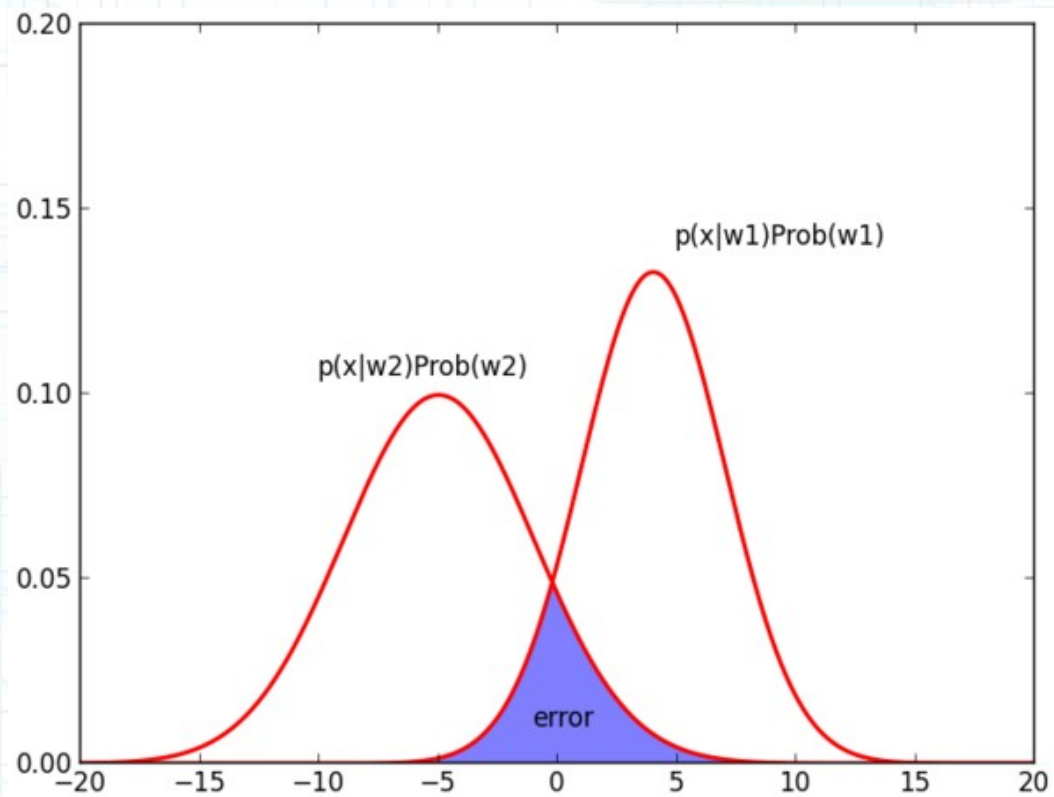
$$LOO(k, \beta_k) \rightarrow \min_{k, \beta_k}$$

3) Будем добавлять признаки, пока LOO уменьшается

Выбор метрики – сложная задача

- Юриспруденция: поиск похожих случаев из судебной практики
- Медицина: как сказывалось то или иное лечение на пациентах с похожими симптомами?
- Оценка стоимости: найти похожие квартиры/поддержанную технику/драгоценности и вычислить среднюю стоимость
- Геномика: найти общие признаки генных последовательностей, отвечающие за диагноз

Вероятностный подход



Вероятностная постановка задачи

- $P(x, y)$ – неизвестная точная плотность распределения на $X \times Y$
- x^ℓ - выборка из случайных, независимых и одинаково распределенных прецедентов
- Найти: эмпирическую оценку плотности
- Классификатор с минимальной вероятностью ошибки:

$$a(x) = \arg \max_{y \in Y} P(y|x) = \arg \max_{y \in Y} P(y)p(x|y)$$

- Классификатор с минимальным средним риском:

$$a(x) = \arg \min_s E_y \mathcal{L}(s, y)$$

Decision function

- Предположим, что мы нашли вероятность $p(y|x) = p(x,y)/p(x)$. Какое значение y нужно предсказать для заданного x ?

- Минимизация среднего риска:

$$a(x) = \arg \min_s E_y \mathcal{L}(s, y)$$

- Упражнение:

y	2	3	4	5
$p(y x)$	0.1	0.2	0.3	0.4

примите правильные решения

$a(x)$ для каждой функции потерь

- бинарная

- $\mathcal{L}(a(x), y^*(x)) = |a(x) - y^*(x)|$

- $\mathcal{L}(a(x), y^*(x)) = (a(x) - y^*(x))^2$

Вероятностные подходы

- Фреквентистский — оценка вероятностного распределения данных:
 - дискриминативный подход, оценивает $p(y|x)$ и строит разделяющую классы поверхность (пример: логистич. регр.)
 - генеративный, оценивает $p(x|y)$ и применяет формулу Байеса

- непараметрический $\hat{p}(x) = \sum_{i=1}^{\ell} w_i K\left(\frac{\rho(x, x_i)}{h}\right)$
- параметрический $\hat{p}(x) = \varphi(x, \theta)$

- Байесовский — оценка случайных параметров модели, данные считаются неслучайными

Наивный байесовский классификатор

- Восстановление n одномерных плотностей — намного более простая задача, чем одной n -мерной.
- Допущение (наивное): признаки являются независимыми случайными величинами
- Тогда совместная плотность распределения представима в виде произведения частных плотностей:
$$p(x|y) = p_1(\xi_1|y) \cdots p_n(\xi_n|y), \quad x = (\xi_1, \dots, \xi_n), \quad y \in Y.$$

Непараметрическая оценка

- Определение плотности вероятности (одномерный случай):

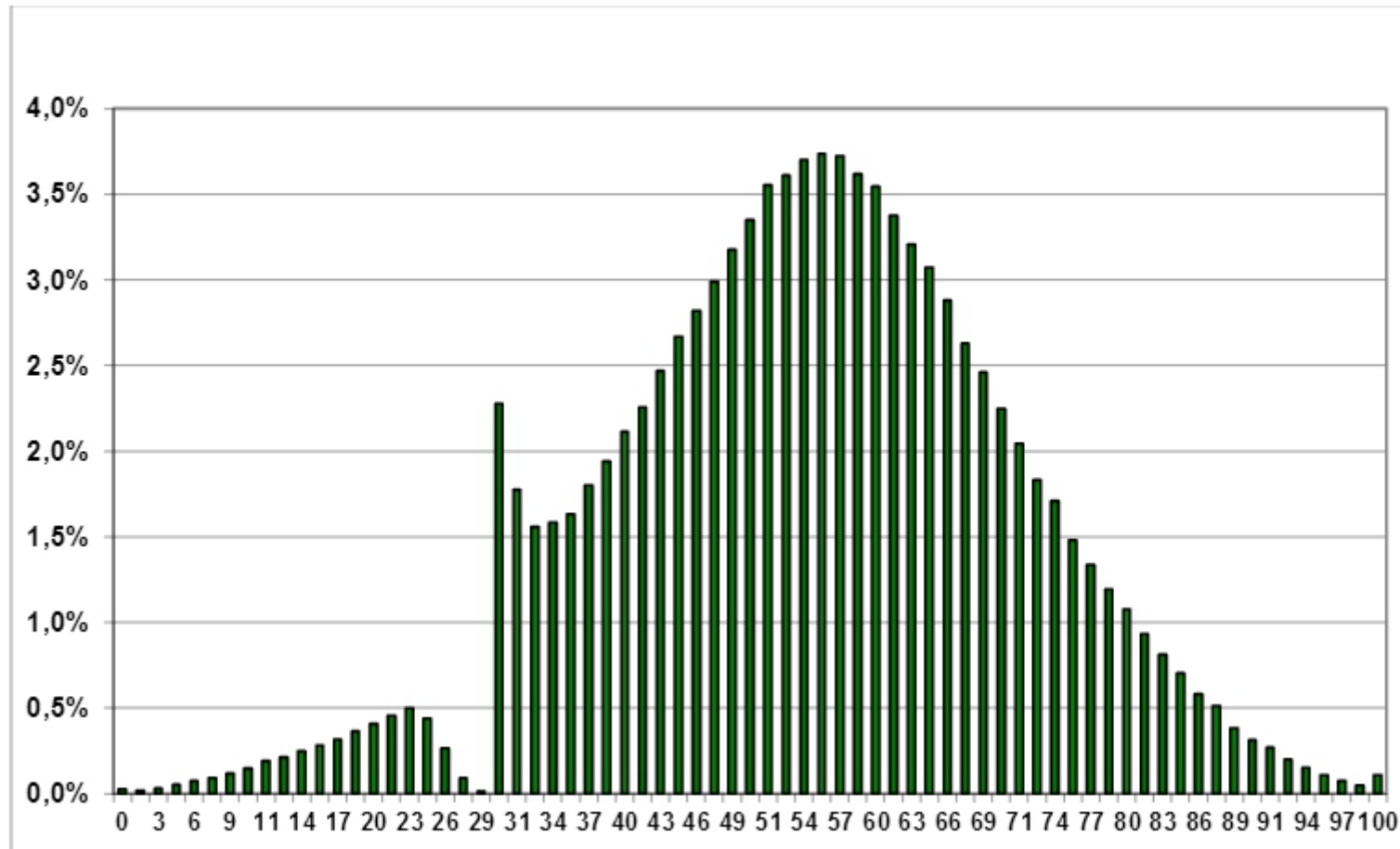
$$p(x) = \lim_{h \rightarrow 0} \frac{1}{2h} P[x - h, x + h]$$

- Эмпирическая оценка:

$$\hat{p}_h(x) = \frac{1}{2h} \frac{1}{\ell} \sum_{i=1}^{\ell} [|x - x_i| < h]$$

Пример – гистограмма оценок

2.1. Poziom podstawowy

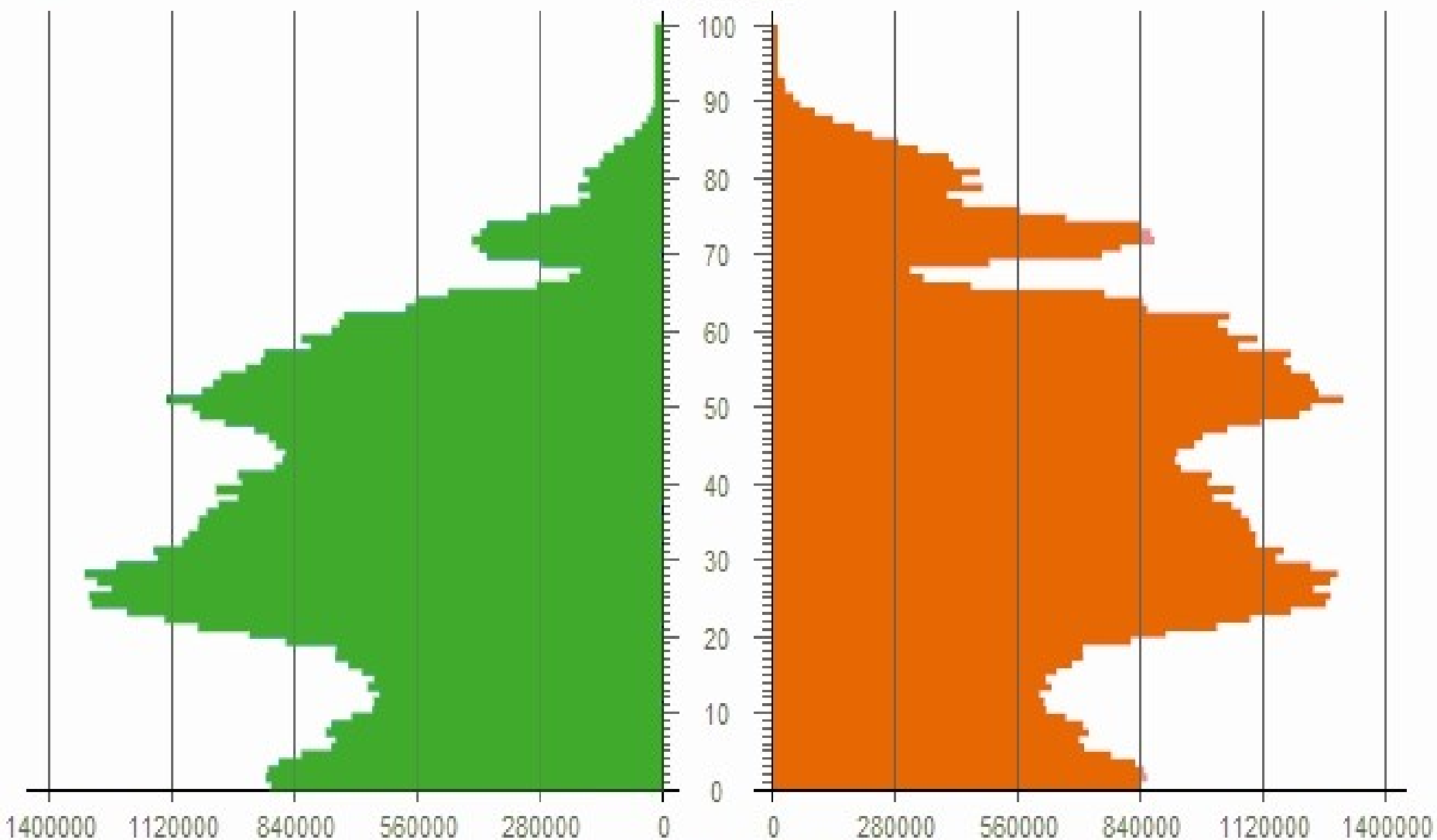


Пример – гистограмма возрастов (Россия 2012г)

Мужчины

Женщины

Возраст (лет)



Численность населения соответствующего возраста, человек

Локальная непараметрическая оценка Парзена-Розенблатта

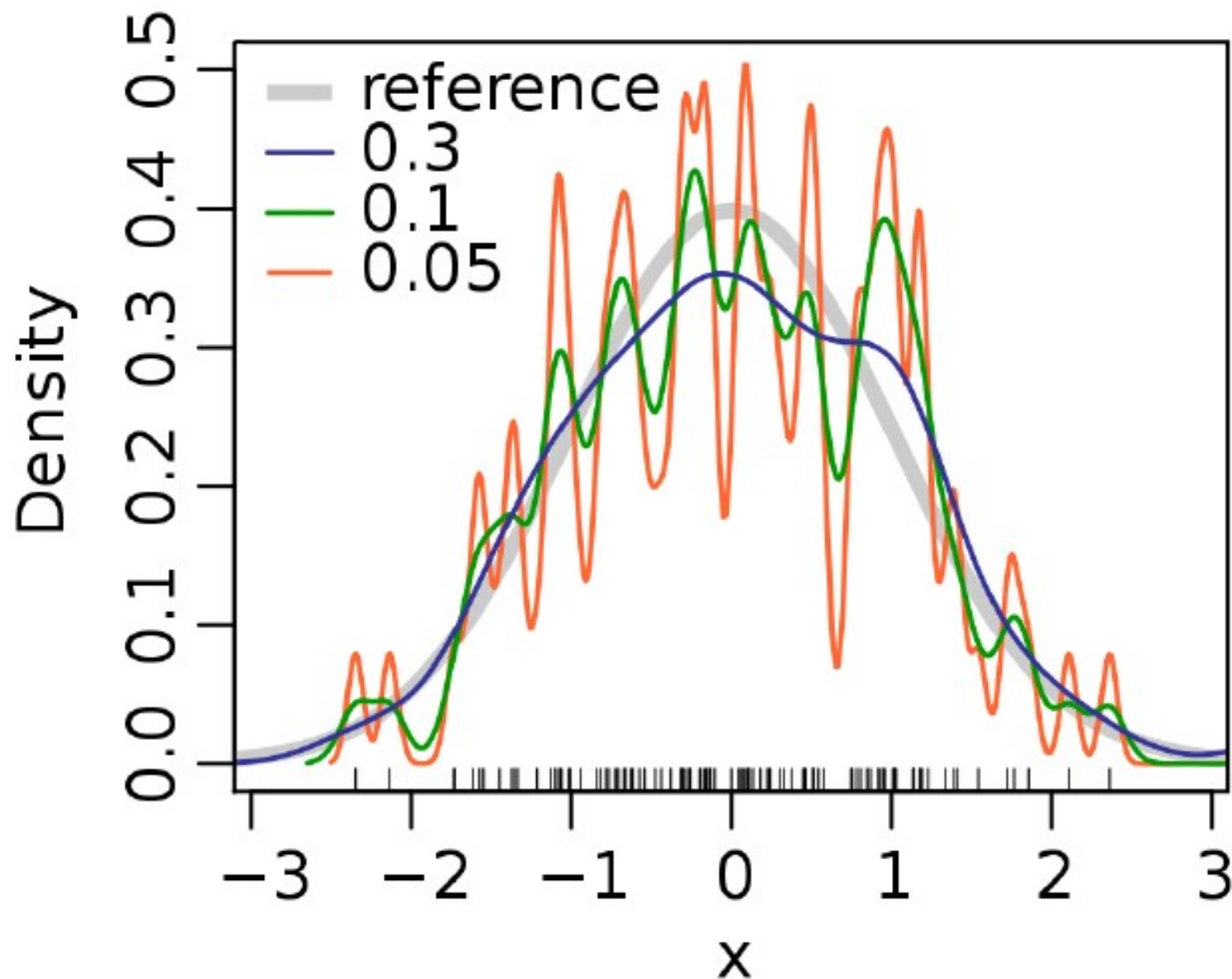
$$\hat{p}_h(x) = \frac{1}{\ell h} \sum_{i=1}^{\ell} K\left(\frac{x - x_i}{h}\right)$$

$K(z)$ — функция, называемая ядром, чётная и нормированная:

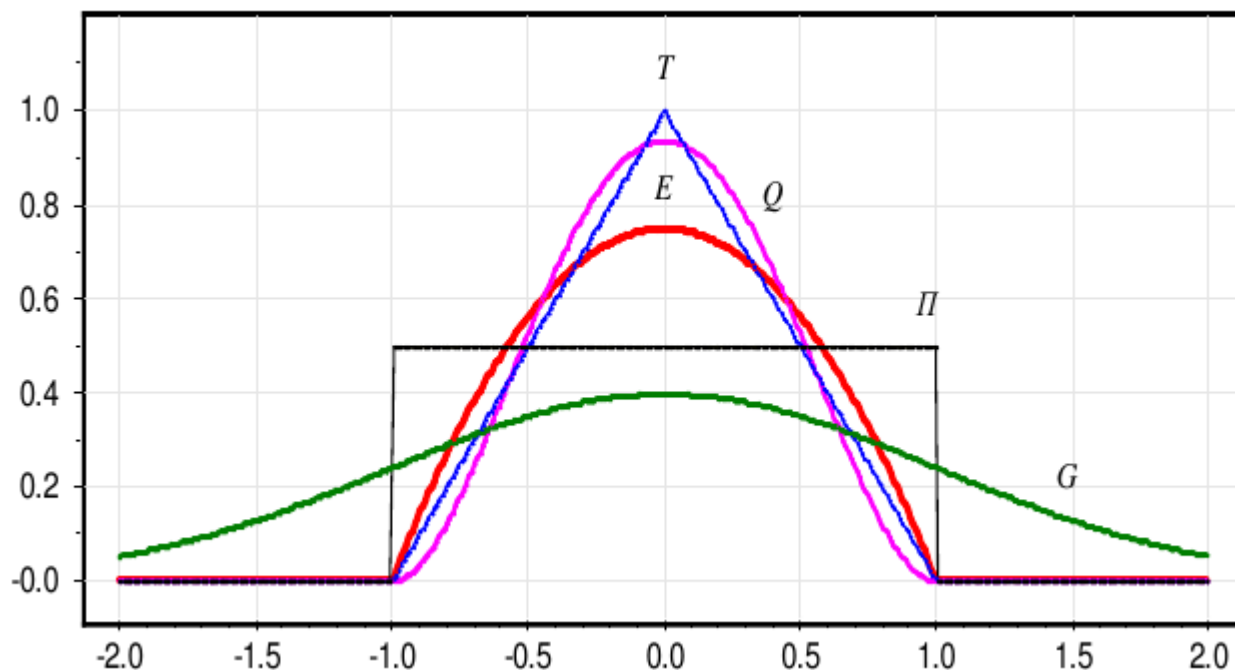
$$\int K(z) dz = 1$$

\hat{p}_h сходится к p при $h \rightarrow 0$, $\ell \rightarrow \infty$, $h\ell \rightarrow \infty$

Зависимость от h



Выбор ядра



$E(r) = \frac{3}{4}(1 - r^2) [|r| \leq 1]$ — оптимальное (Епанечникова);

$Q(r) = \frac{15}{16}(1 - r^2)^2 [|r| \leq 1]$ — четвертое;

$T(r) = (1 - |r|) [|r| \leq 1]$ — треугольное;

$G(r) = (2\pi)^{-1/2} \exp(-\frac{1}{2}r^2)$ — гауссовское;

$\Pi(r) = \frac{1}{2} [|r| \leq 1]$ — прямоугольное.

Параметрическая оценка плотности

$$p(x) = \varphi(x; \theta)$$

- Принцип максимума правдоподобия:

$$L(\theta; X^\ell) = \sum_{i=1}^{\ell} \ln \varphi(x_i; \theta) \rightarrow \max_{\theta}$$

- Необходимое условие оптимума:

$$\frac{\partial}{\partial \theta} L(\theta; X^\ell) = \sum_{i=1}^{\ell} \frac{\partial}{\partial \theta} \ln \varphi(x_i; \theta) = 0$$

Многомерное нормальное распределение

$$a(x) = \arg \max_{y \in Y} P(y|x) = \arg \max_{y \in Y} P(y)p(x|y)$$

$$p(x|y) = \mathcal{N}(x; \mu_y, \Sigma_y) = \frac{e^{-\frac{1}{2}(x-\mu_y)^\top \Sigma_y^{-1}(x-\mu_y)}}{\sqrt{(2\pi)^n \det \Sigma_y}}$$

где $\mu_y \in \mathbb{R}^n$ — вектор математического ожидания (центр) класса $y \in Y$
 $\Sigma_y \in \mathbb{R}^{n \times n}$ — ковариационная матрица класса $y \in Y$

Принцип максимума правдоподобия:

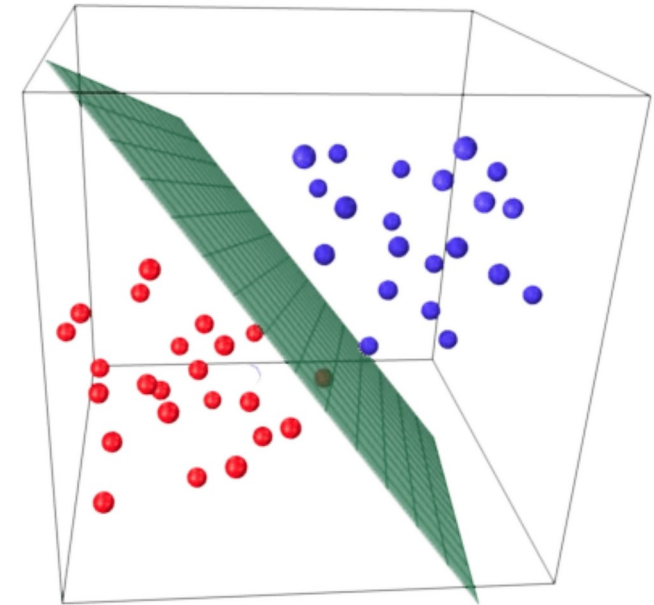
$$L(\theta, X^\ell) = \prod_{i=1}^{\ell} \varphi(x_i, y_i, \theta) \rightarrow \max_{\theta}$$

Решение – подстановочный алгоритм:

$$\hat{\mu} = \frac{1}{\ell} \sum_{i=1}^{\ell} x_i; \quad \hat{\Sigma} = \frac{1}{\ell} \sum_{i=1}^{\ell} (x_i - \hat{\mu})(x_i - \hat{\mu})^\top$$

Логистическая регрессия

Если плотности вероятностей объектов в каждом классе распределены по нормальному закону с одинаковой ковариационной матрицей, но разными математическими ожиданиями, то разделяющая классы поверхность является плоскостью, а вероятности равны логистической функции от отклонения точек от плоскости

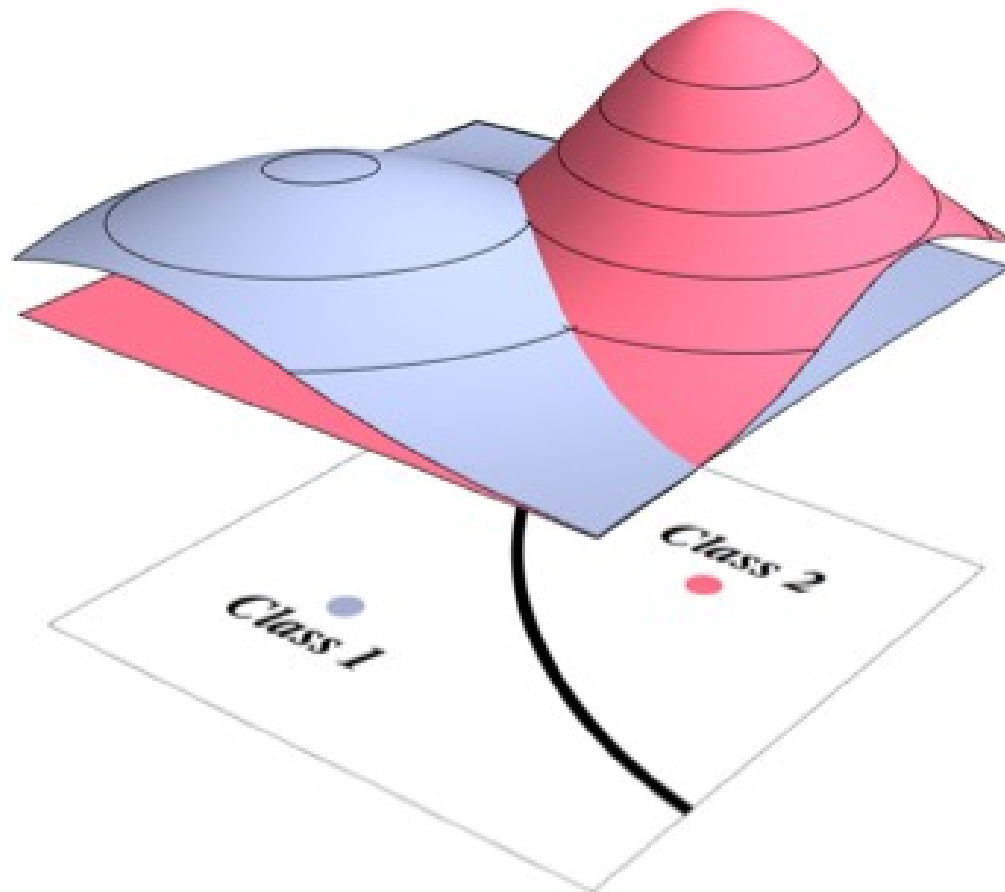


$$a(x) = \text{sign}(\langle w, x \rangle - w_0)$$

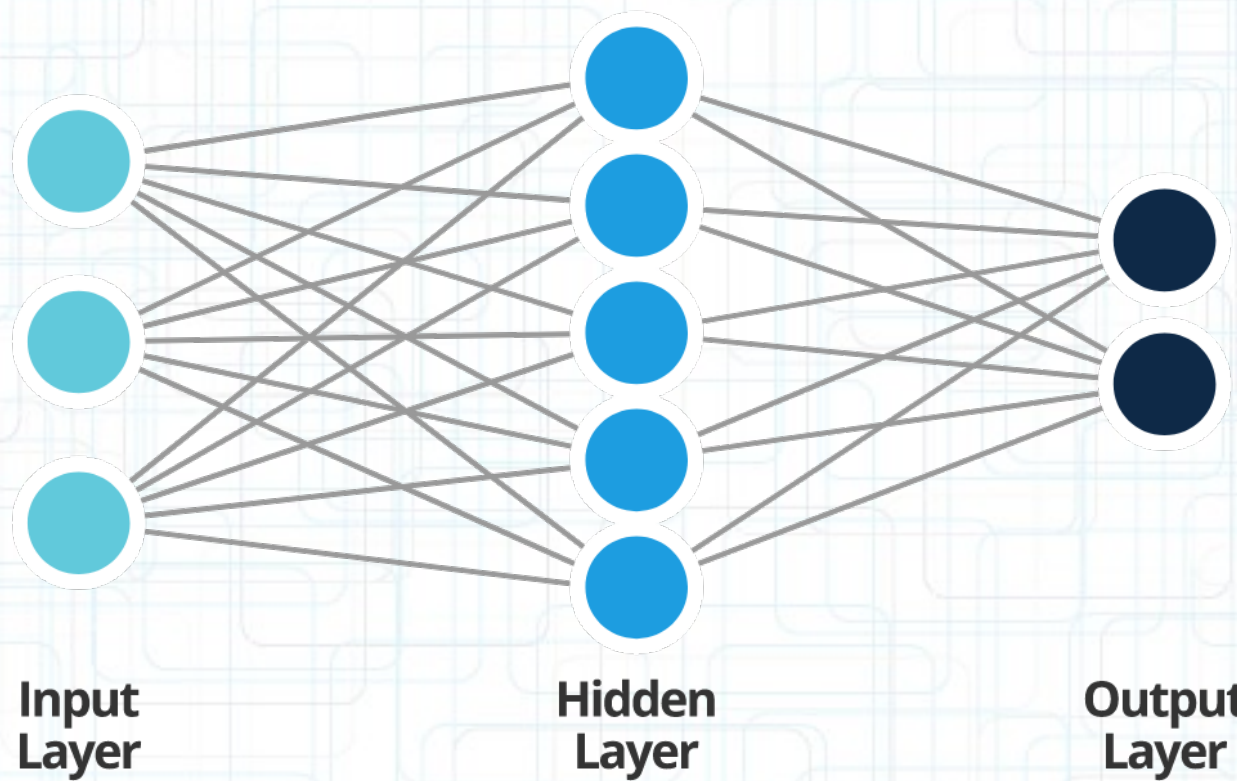
$$P(y|x) = \sigma(\langle w, x \rangle y)$$

$$\sigma(z) = \frac{1}{1+e^{-z}}$$

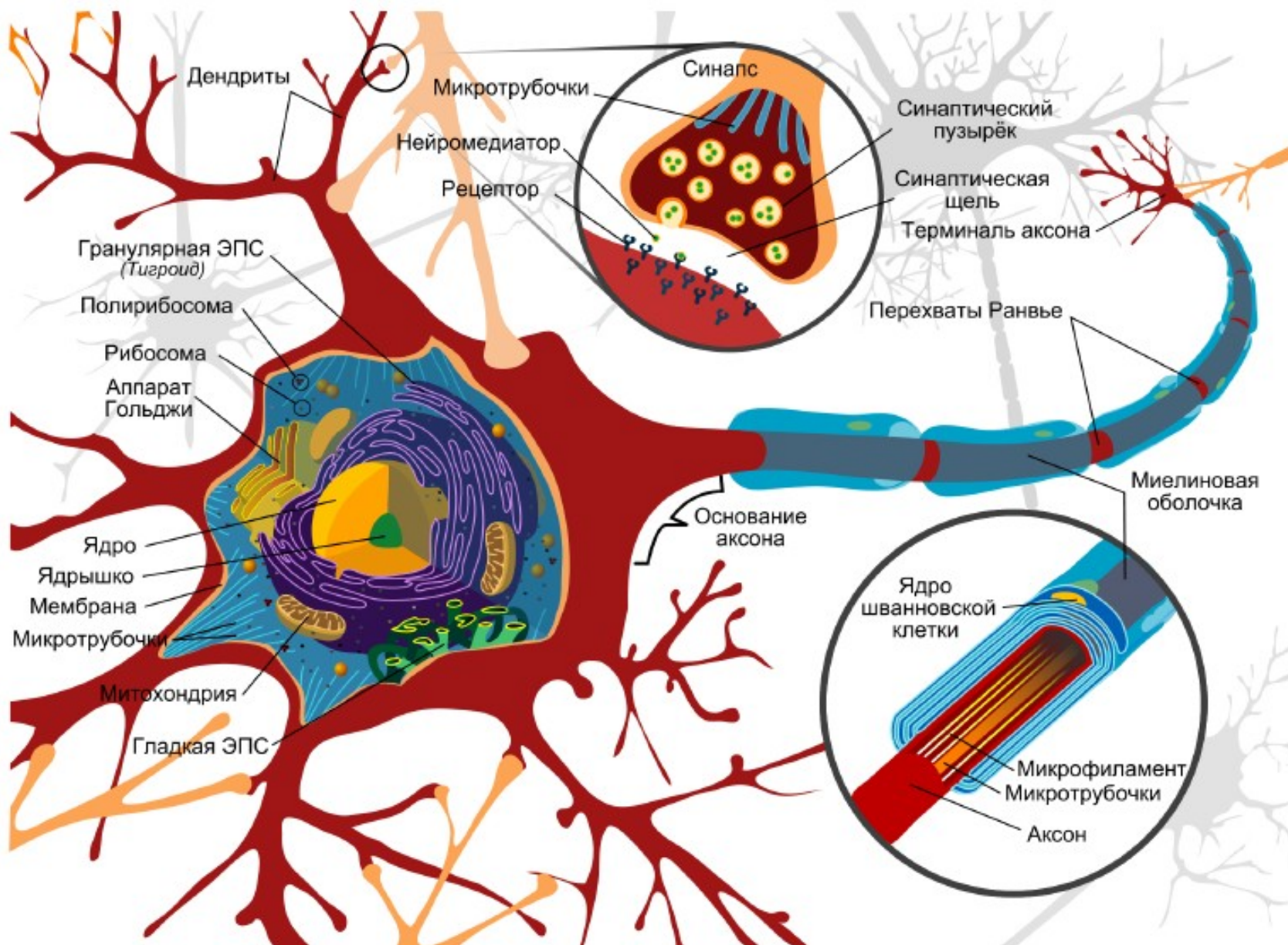
Разные дисперсии приводят к нелинейной разделяющей классы поверхности



Нейросетевые алгоритмы

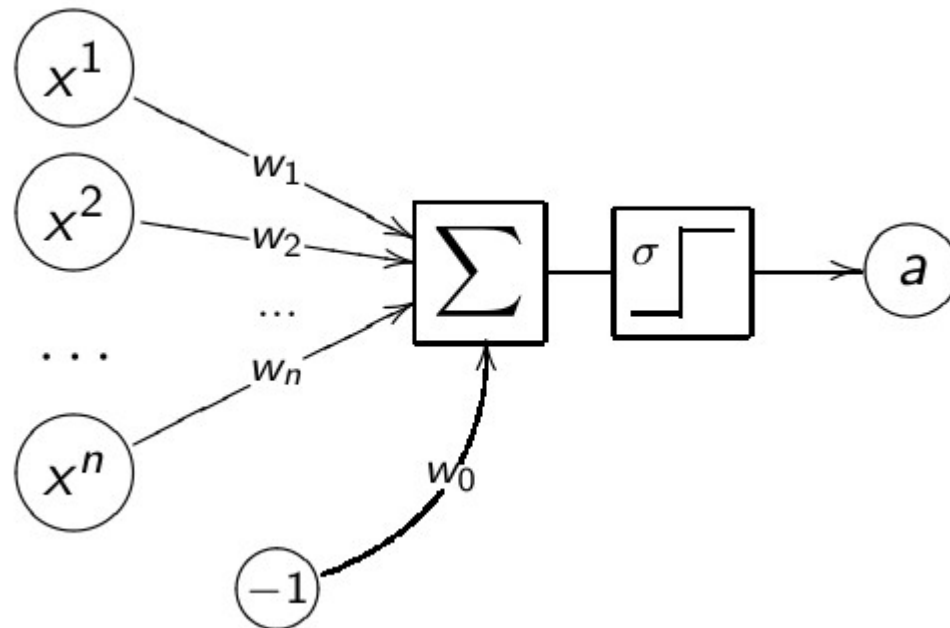


Модель нейрона



Линейная модель нейрона МакКаллока-Питтса (1943)

$$a(x, w) = \sigma(\langle w, x \rangle) = \sigma\left(\sum_{j=1}^n w_j f_j(x) - w_0\right)$$



Градиентный метод численной минимизации

Минимизация эмпирического риска:

$$Q(w) = \sum_{i=1}^{\ell} \mathcal{L}(g(w, x_i), y_i) = \sum_{i=1}^{\ell} \mathcal{L}_i(w) \rightarrow \min_w.$$

Численная минимизация методом *градиентного спуска*:

$w^{(0)}$:= начальное приближение;

$$w^{(t+1)} := w^{(t)} - h \cdot \nabla Q(w^{(t)}), \quad \nabla Q(w) = \left(\frac{\partial Q(w)}{\partial w_j} \right)_{j=0}^n,$$

где h — *градиентный шаг*, называемый также *темпом обучения*.

$$w^{(t+1)} := w^{(t)} - h \sum_{i=1}^{\ell} \nabla \mathcal{L}_i(w^{(t)}).$$

Идея ускорения сходимости:

брать (x_i, y_i) по одному и сразу обновлять вектор весов.

Достоинства и недостатки

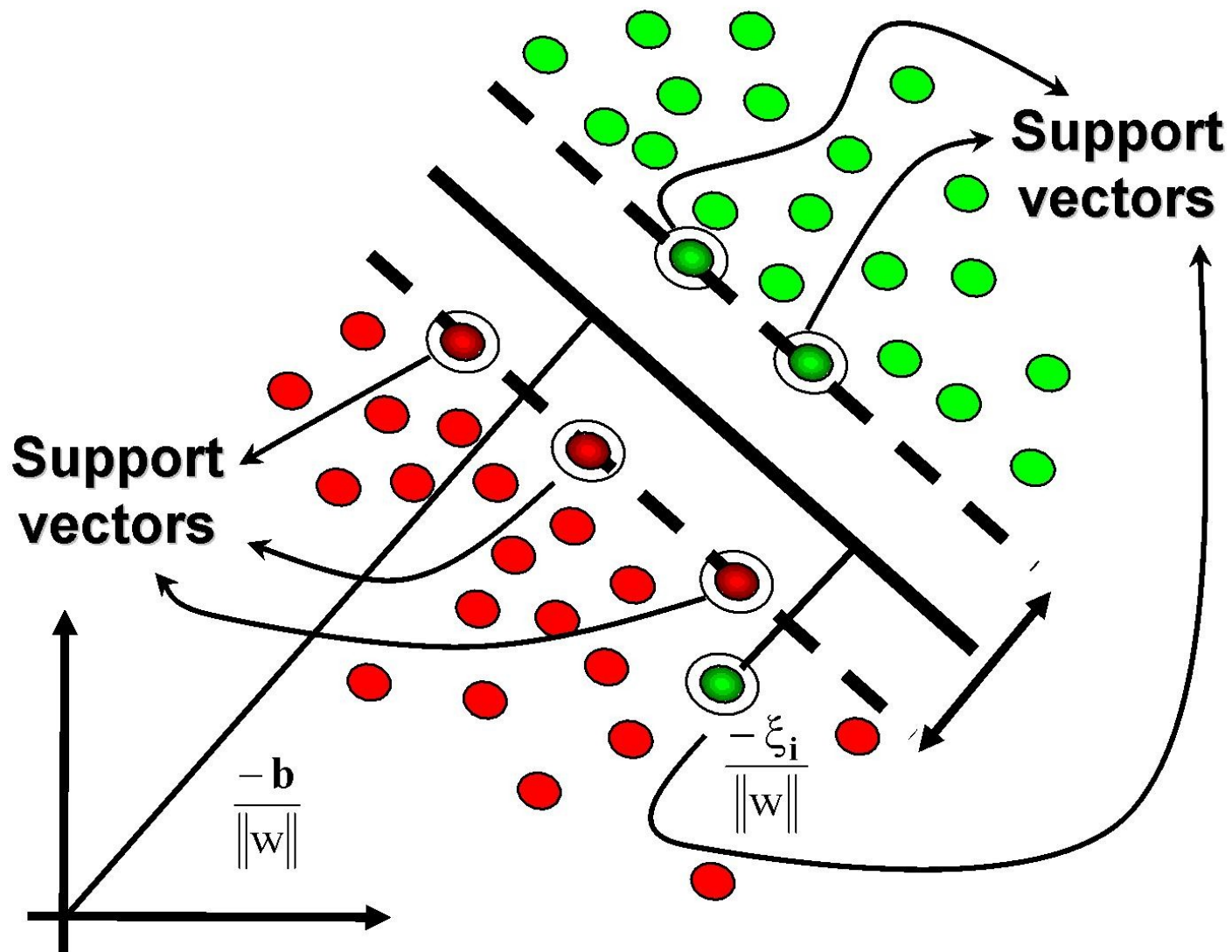
Достоинства:

- 1 легко реализуется;
- 2 легко обобщается на любые $g(x, w)$, $\mathcal{L}(a, y)$;
- 3 возможно динамическое (потокковое) обучение;
- 4 на сверхбольших выборках можно получить неплохое решение, даже не обработав все (x_i, y_i) ;
- 5 всё чаще применяется для Big Data

Недостатки:

- 1 возможна расходимость или медленная сходимость;
- 2 застревание в локальных минимумах;
- 3 подбор комплекса эвристик является искусством;
- 4 проблема переобучения;

Метод опорных векторов (SVM)



Самая широкая разделяющая полоса

- Рассмотрим линейный классификатор:

$$a(x, w) = \text{sign}(\langle w, x \rangle - w_0)$$

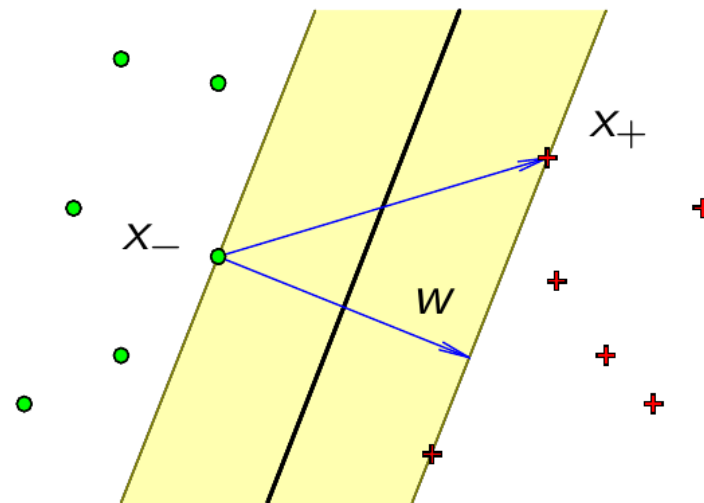
- Допустим, что обучающая выборка линейно разделима:

$$\exists w, w_0 : M_i(w, w_0) = y_i(\langle w, x_i \rangle - w_0) > 0, \quad i = 1, \dots, \ell$$

- w и w_0 определены с точностью до множителя \Rightarrow нормируем $\min_{i=1, \dots, \ell} M_i(w, w_0) = 1$

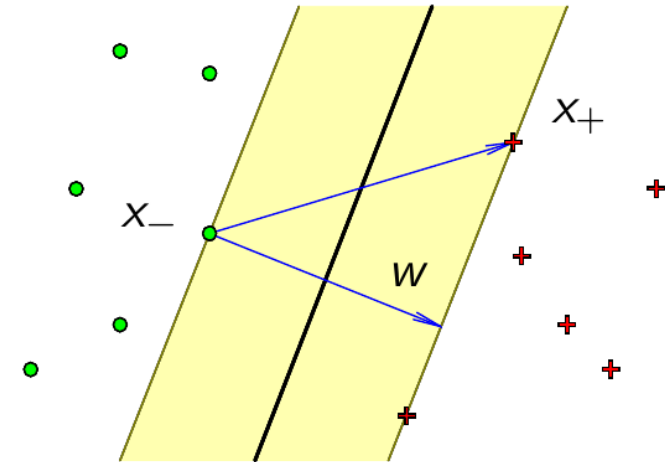
- Ширина полосы:

$$\frac{\langle x_+ - x_-, w \rangle}{\|w\|} = \frac{2}{\|w\|} \rightarrow \max$$



Метод опорных векторов для линейно разделимой выборки

$$\begin{cases} \frac{1}{2} \|w\|^2 \rightarrow \min; \\ M_i(w, w_0) \geq 1, \quad i = 1, \dots, \ell \end{cases}$$



Что делать, если выборка не разделима гиперплоскостью?

Случай линейно неразделимой выборки

$$\begin{cases} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^{\ell} \xi_i \rightarrow \min_{w, w_0, \xi}; \\ M_i(w, w_0) \geq 1 - \xi_i, \quad i = 1, \dots, \ell; \\ \xi_i \geq 0, \quad i = 1, \dots, \ell. \end{cases}$$

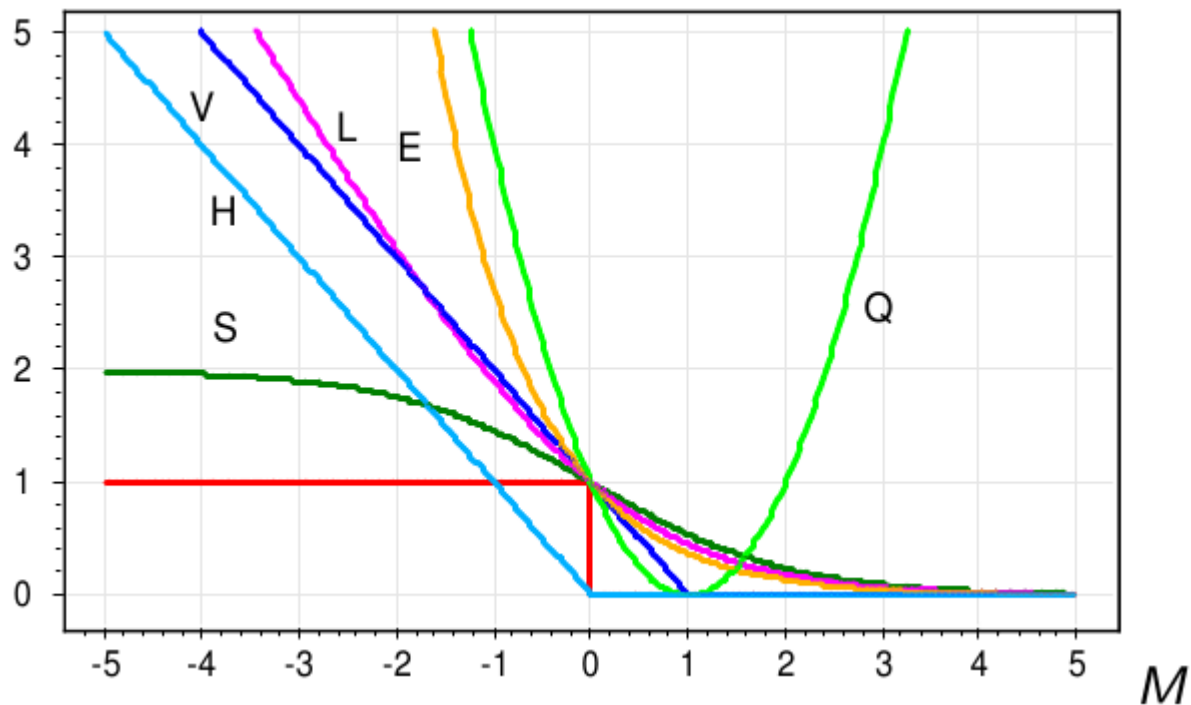
Так как $\xi_i \geq 0$ и $\xi_i \geq 1 - M_i$, то в силу минимизации суммы ξ_i

$$\xi_i = (1 - M_i)_+$$

Следовательно, наша задача эквивалентна минимизации функционала

$$Q(w, w_0) = \sum_{i=1}^{\ell} (1 - M_i(w, w_0))_+ + \frac{1}{2C} \|w\|^2 \rightarrow \min_{w, w_0}$$

Часто используемые функции потерь



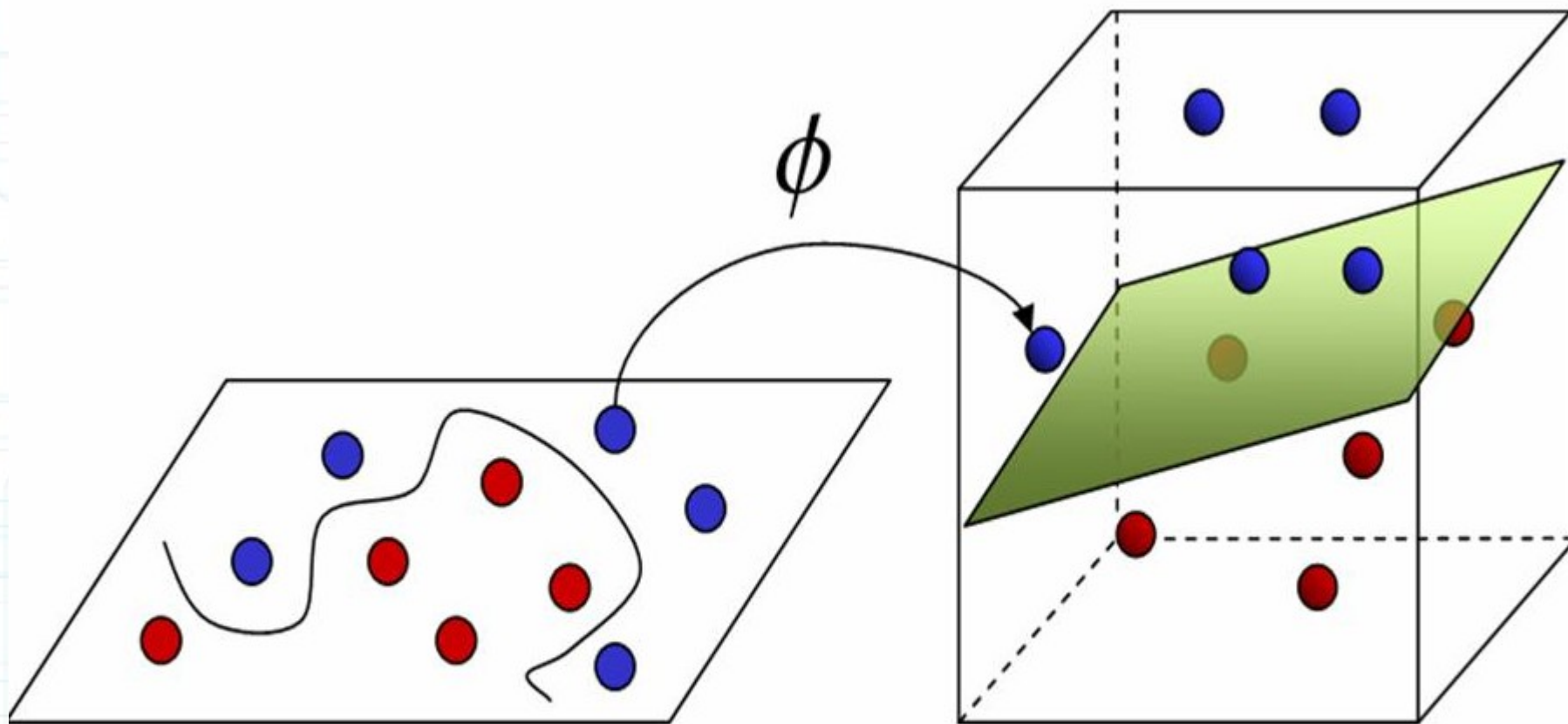
$$V(M) = (1 - M)_+$$
$$H(M) = (-M)_+$$
$$L(M) = \log_2(1 + e^{-M})$$
$$Q(M) = (1 - M)^2$$
$$S(M) = 2(1 + e^M)^{-1}$$
$$E(M) = e^{-M}$$

$[M < 0]$

- кусочно-линейная (SVM);
- кусочно-линейная (Hebb's rule);
- логарифмическая (LR);
- квадратичная (FLD);
- сигмоидная (ANN);
- экспоненциальная (AdaBoost);
- пороговая функция потерь.

Нелинейное обобщение SVM

Расширение пространства

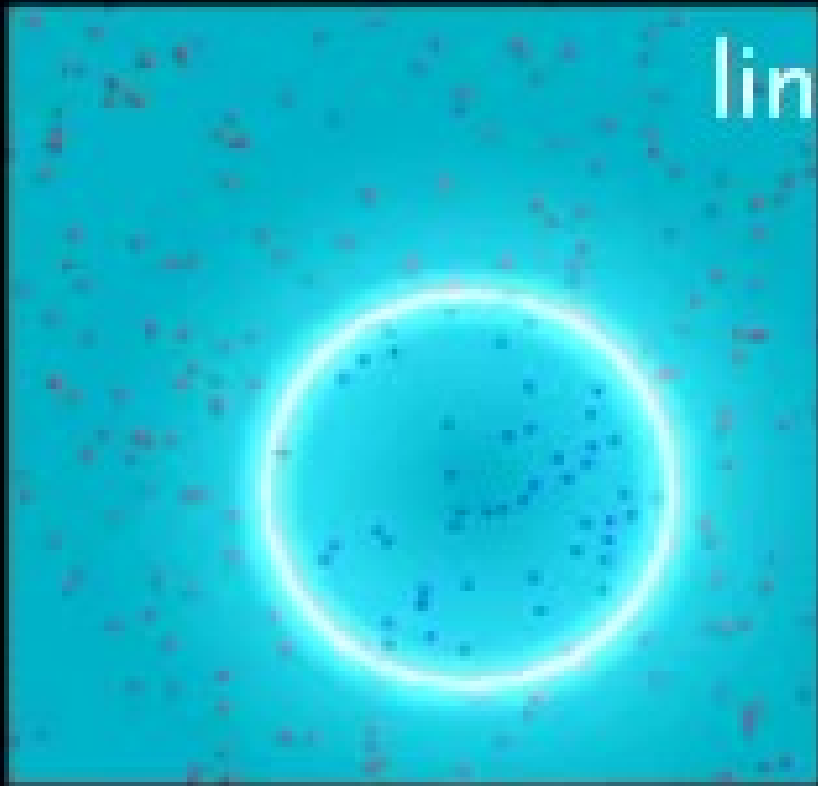


Input Space

Feature Space

Видео-демонстрация

The blue/red
dots are not
linearly separable



Полиномиальные ядра

Расширение пространства $\psi: (u_1, u_2) \mapsto (u_1^2, u_2^2, \sqrt{2}u_1u_2)$

$$K(x, x') = \langle \psi(x), \psi(x') \rangle_H$$

Эквивалентно введению нового скалярного произведения в исходном пространстве:

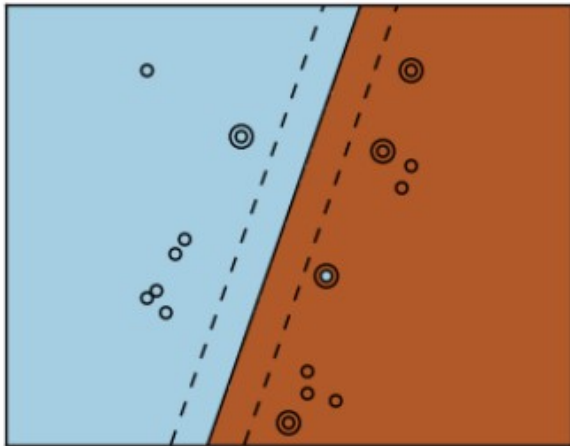
$$\begin{aligned} K(u, v) &= \langle u, v \rangle^2 = \langle (u_1, u_2), (v_1, v_2) \rangle^2 = \\ &= (u_1v_1 + u_2v_2)^2 = u_1^2v_1^2 + u_2^2v_2^2 + 2u_1v_1u_2v_2 = \\ &= \langle (u_1^2, u_2^2, \sqrt{2}u_1u_2), (v_1^2, v_2^2, \sqrt{2}v_1v_2) \rangle. \end{aligned}$$

В общем случае новое скалярное произведение вводится формулой:

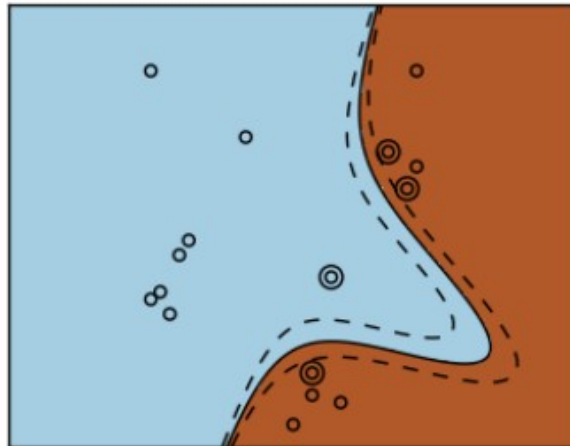
$$K(x, x') = (\langle x, x' \rangle + 1)^d$$

Примеры классификаций с различными ядрами

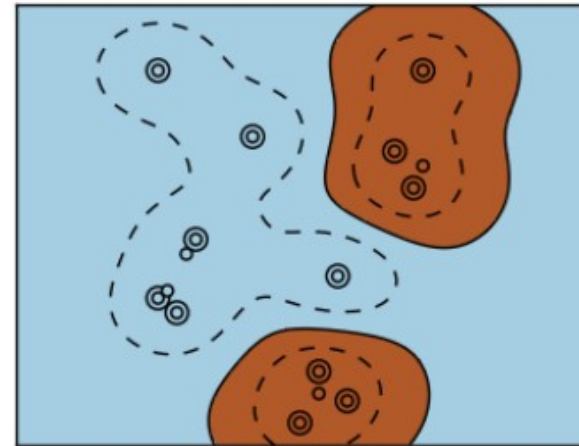
линейное
 $\langle x, x' \rangle$



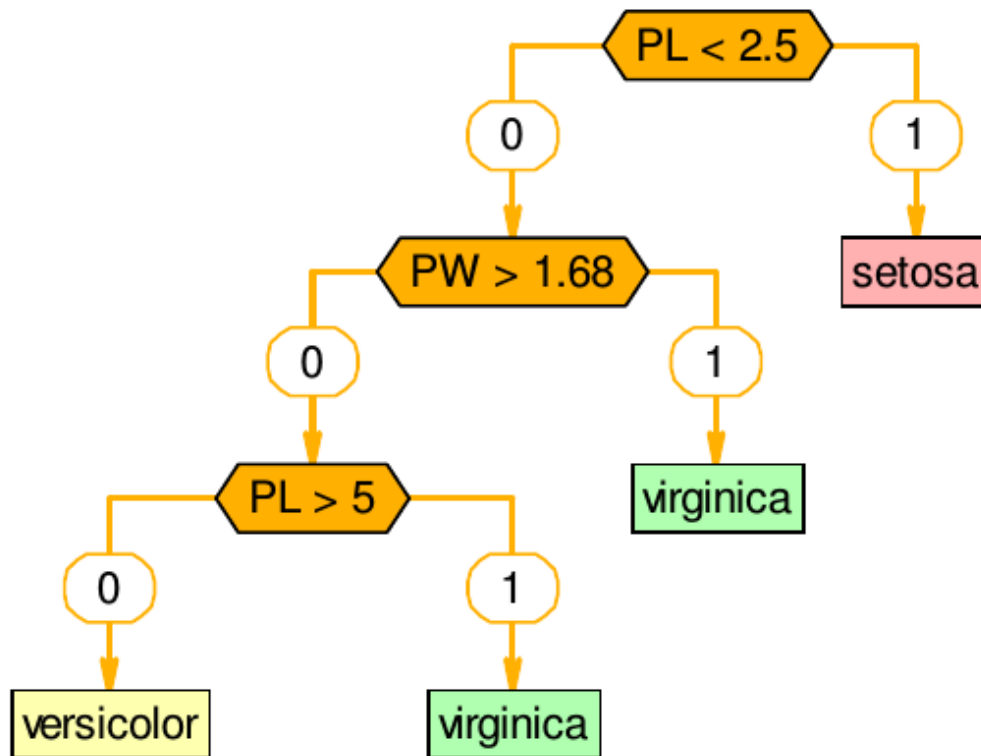
полиномиальное
 $(\langle x, x' \rangle + 1)^d, d=3$



гауссовское (RBF)
 $\exp(-\beta \|x - x'\|^2)$



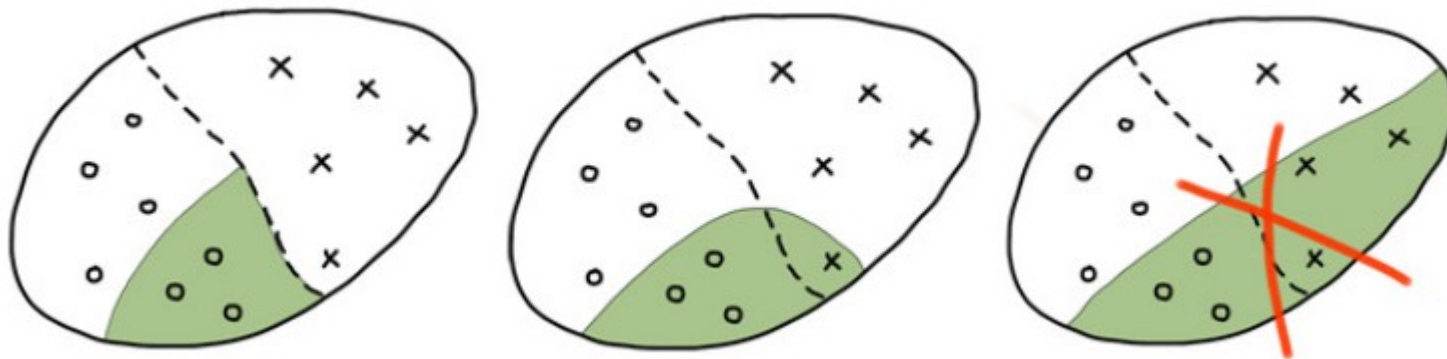
Логические алгоритмы



Понятие закономерности

- Предикат $R: X \rightarrow \{0, 1\}$ – закономерность, если он выделяет ($R(x)=1$) достаточно много объектов одного класса C и практически не выделяет объектов других классов

$$p_c(R) = \#\{x_i : R(x_i)=1 \text{ и } y_i=c\} \rightarrow \max;$$
$$n_c(R) = \#\{x_i : R(x_i)=1 \text{ и } y_i \neq c\} \rightarrow \min;$$



Точный тест Фишера

- Предположим, что события “объект отобран предикатом” и “объект имеет класс c ” независимы
- Тогда вероятность отобрать r объектов класса c и $n - r$ – других классов:

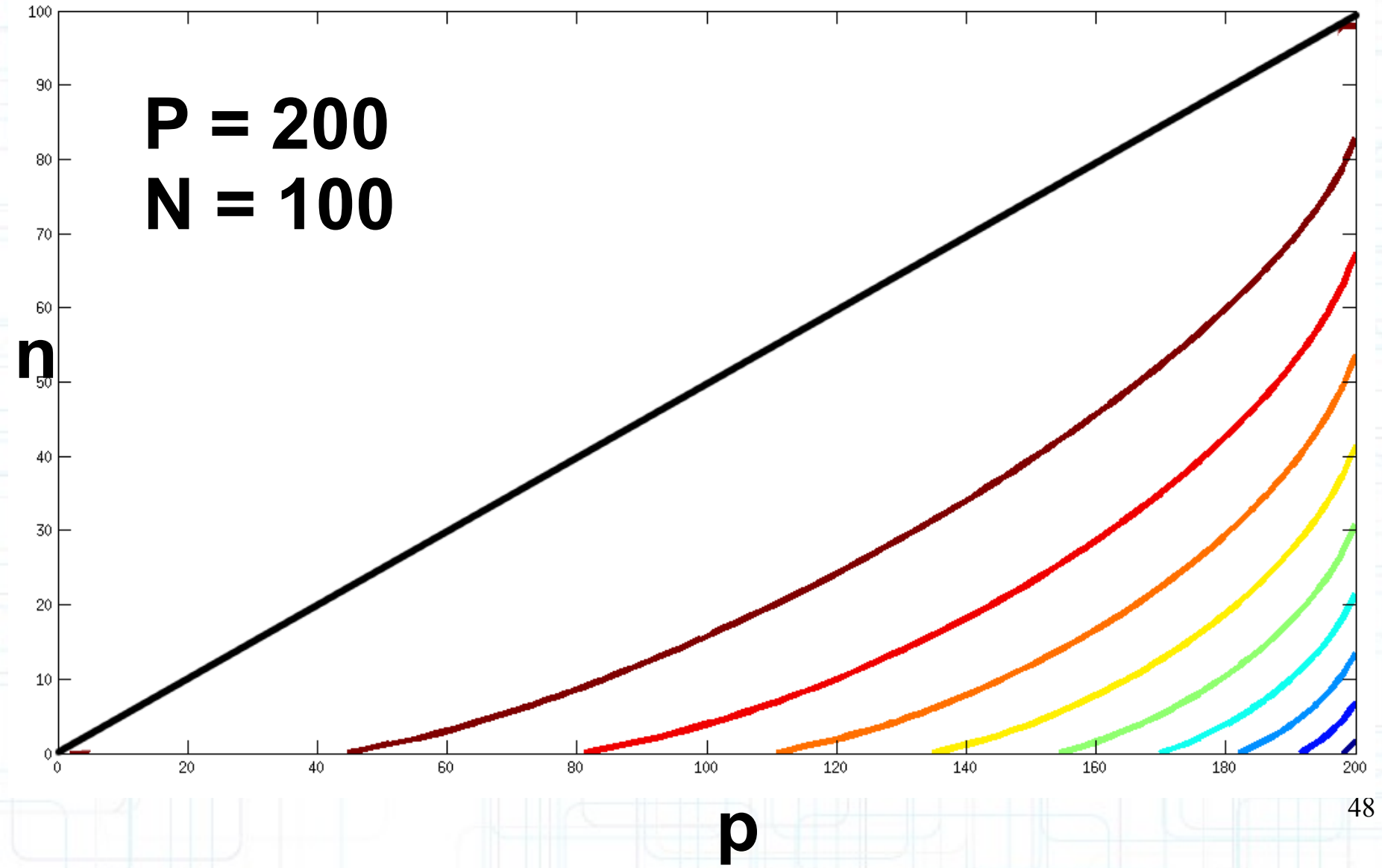
Точный тест Фишера

- Предположим, что события “объект отобран предикатом” и “объект имеет класс c ” независимы
- Тогда вероятность отобрать p объектов класса c и n – других классов: $\frac{C_P^p C_N^n}{C_{P+N}^{p+n}}$
- Это правдоподобие гипотезы независимости событий. Чем меньше данная вероятность, тем более зависимы события

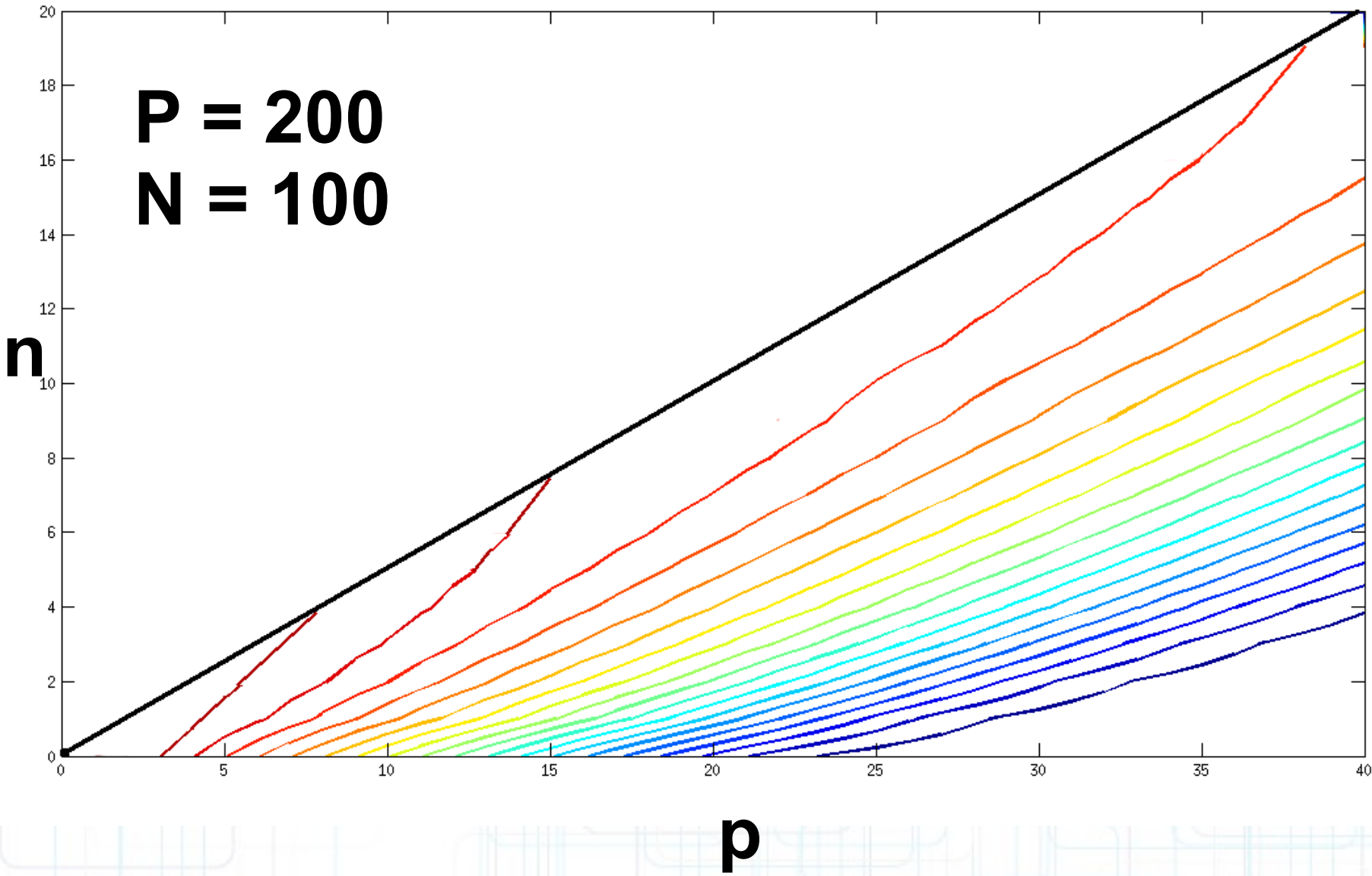
$$IStat(p, n) = -\frac{1}{\ell} \log_2 \frac{C_P^p C_N^n}{C_{P+N}^{p+n}} \rightarrow \max$$

Линии уровня теста Фишера Малые p и n

$P = 200$
 $N = 100$



Линии уровня теста Фишера Малые p и n



Энтропийный критерий информативности

Пусть ω_0, ω_1 — два исхода с вероятностями q и $1 - q$.

Количество информации: $I_0 = -\log_2 q$, $I_1 = -\log_2(1 - q)$.

Энтропия — математическое ожидание количества информации:

$$h(q) = -q \log_2 q - (1 - q) \log_2(1 - q).$$

Энтропия выборки X^ℓ , если исходы — это классы $y=c$, $y \neq c$:

$$H(y) = h\left(\frac{P}{\ell}\right).$$

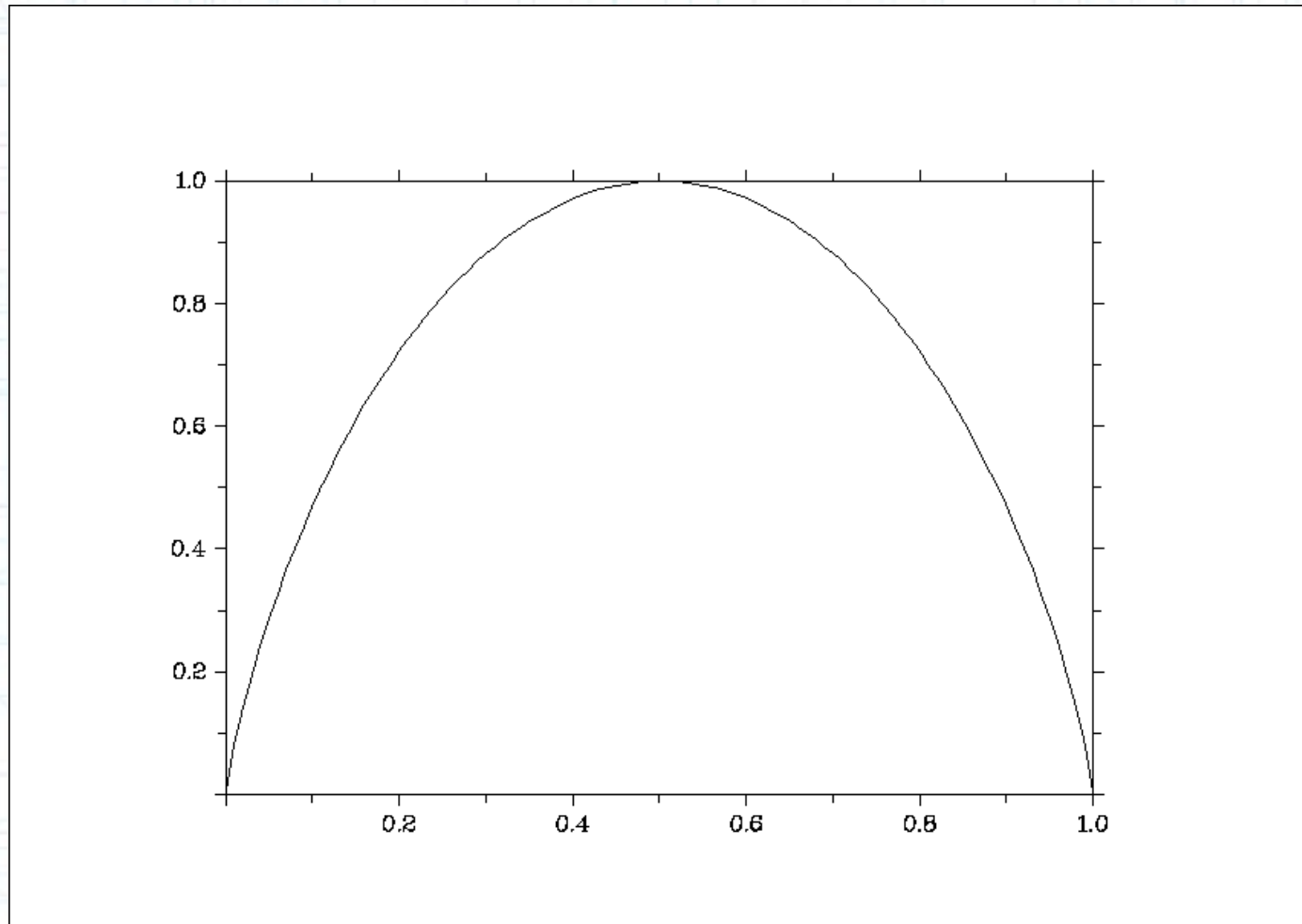
Энтропия выборки X^ℓ после получения информации $R(x_i)_{i=1}^\ell$:

$$H(y|R) = \frac{p+n}{\ell} h\left(\frac{p}{p+n}\right) + \frac{\ell-p-n}{\ell} h\left(\frac{P-p}{\ell-p-n}\right).$$

Прирост информации (Information gain, IGain):

$$\text{IGain}(p, n) = H(y) - H(y|R).$$

Энтропия для различных q



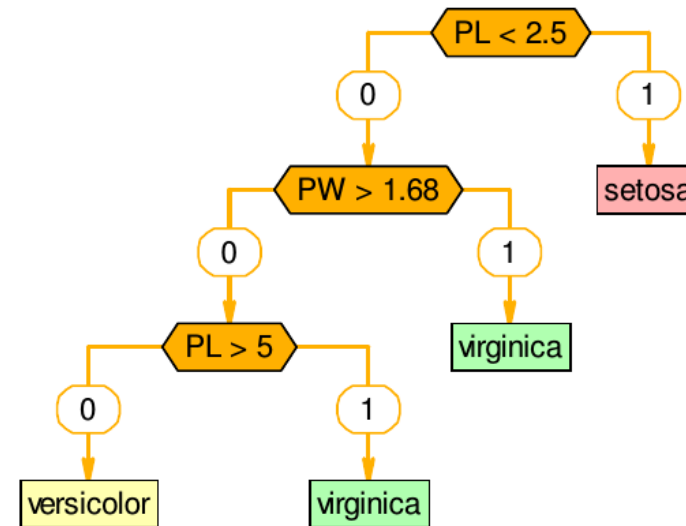
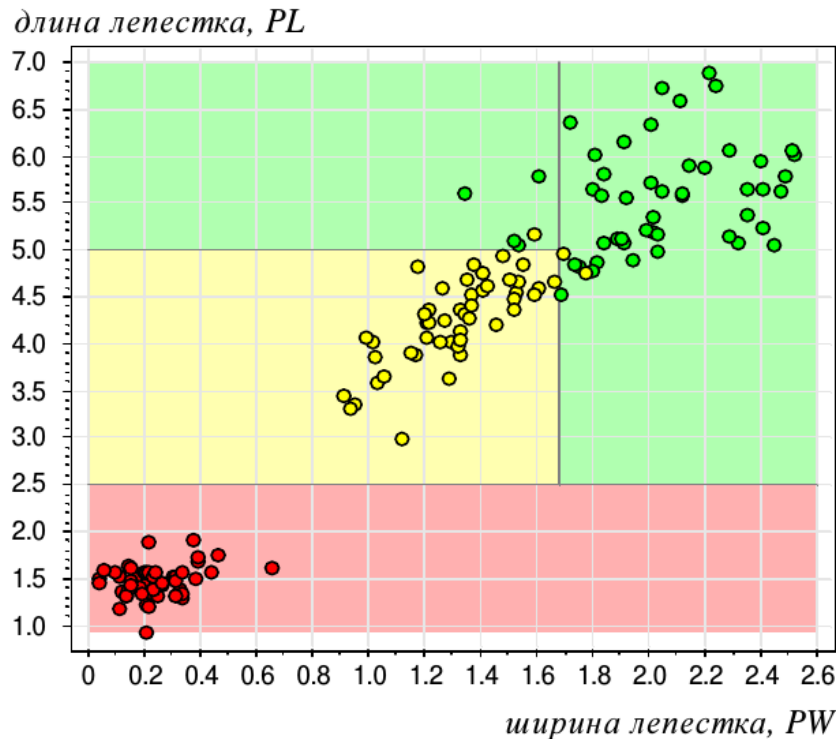
Соотношение статистического и энтропийного критериев

Энтропийный критерий $I\text{Gain}$ асимптотически эквивалентен статистическому $I\text{Stat}$:

$$I\text{Stat}(p, n) \rightarrow I\text{Gain}(p, n) \quad \text{при } \ell \rightarrow \infty$$

Доказательство: применить формулу Стирлинга к критерию $I\text{Stat}$.

Решающее дерево → покрывающий набор конъюнкций



setosa	$r_1(x) = [PL \leq 2.5]$
virginica	$r_2(x) = [PL > 2.5] \wedge [PW > 1.68]$
virginica	$r_3(x) = [PL > 5] \wedge [PW \leq 1.68]$
versicolor	$r_4(x) = [PL > 2.5] \wedge [PL \leq 5] \wedge [PW < 1.68]$

Жадный алгоритм построения решающего дерева

- Функция:
- Tree buildTree(U) {
 - Выбор предиката $\beta_v: I(\beta_v, U) \rightarrow \max$
 - $U_0 := \{x \in U \mid \beta_v(x) = 0\}$
 - $U_1 := \{x \in U \mid \beta_v(x) = 1\}$
 - Если $|U_0| < \ell_0$ или $|U_1| < \ell_0$ вернуть лист
 - Иначе:
 - $L_v := \text{buildTree}(U_0)$
 - $R_v := \text{buildTree}(U_1)$
- }

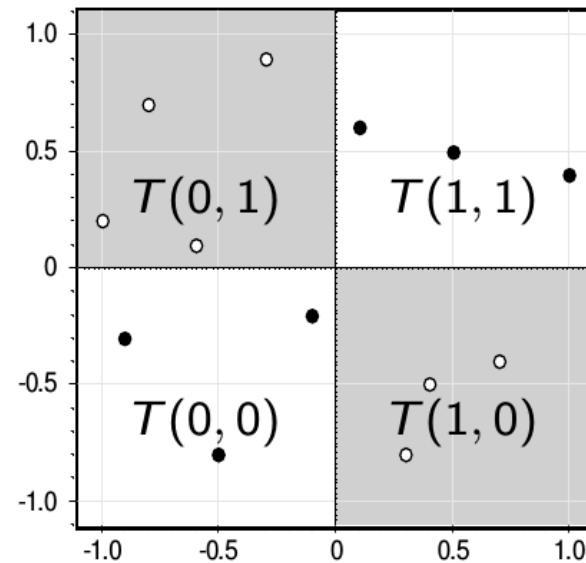
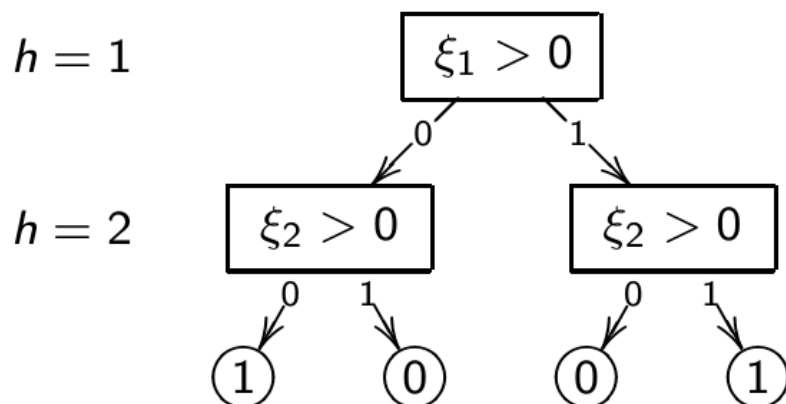
Обобщение на случай задачи регрессии

- В каждом листе целевое значение определяется по методу наименьших квадратов
- Критерий информативности – среднеквадратическая ошибка

Небрежные решающие деревья (Oblivious Decision Tree)

- Для всех узлов на глубине h условие ветвления одинаково
- Дерево получается сбалансированным, на глубине h ровно 2^{h-1} вершин

Пример: задача XOR, $H = 2$.



Сравнение алгоритмов

	kNN	Вероятн.	Нейронные	SVM	Деревья (лес)
Качество	не очень	среднее	хорошее	хорошее	хорошее
Трудоемкость настройки	просто	нужно придумать модель	сложно	просто	просто
Переобучение	для малых k	нет	нужны спец. приемы	нет	нет
Большая размерность	проклятие	если нет зависимостей	нужны спец. слои	ОК	ОК
Маленькие выборки	плохо	Фреквентист. - плохо	нужны пред-обученные сети	ОК	ОК для Extra Trees
Интерпретируемость	понятно	понятно	черн. ящик	понятно	понятно
Нужно нормировать признаки?	да, или подбирать метрику	да	да	да	нет
Скорость обучения/предсказания	∞ / медленно	быстро/быстро	медленно/медленно	быстро/быстро	средне/быстро