

Лекция 8: линейные методы

Евгений Борисов

Линейные методы

методы ML

- *метрические* - померять расстояния, определить ближайших
- *логические* - построить правило (комбинацию предикатов)
- *линейные* - построить разделяющую поверхность
- *статистические* - восстановить плотность, определить вероятность
- *композиции* - собрать несколько классификаторов в один

Линейные методы: о задаче классификации

метки классов

$$Y = \{-1, 1\}$$

размеченные данные

$$X = (x, y)$$

Линейные методы: о задаче классификации

метки классов

$$Y = \{-1, 1\}$$

размеченные данные

$$X = (x, y)$$

алгоритм классификации

$$a(x, w) = \text{sign}(f(x, w))$$

Линейные методы: о задаче классификации

метки классов

$$Y = \{-1, 1\}$$

размеченные данные

$$X = (x, y)$$

алгоритм классификации

$$a(x, w) = \text{sign}(f(x, w))$$

дискриминантная функция

$$f(x, w)$$

вектор параметров

$$w$$

Линейные методы: о задаче классификации

метки классов

$$Y = \{-1, 1\}$$

размеченные данные

$$X = (x, y)$$

алгоритм классификации

$$a(x, w) = \text{sign}(f(x, w))$$

дискриминантная функция

$$f(x, w)$$

вектор параметров

$$w$$

разделяющая поверхность

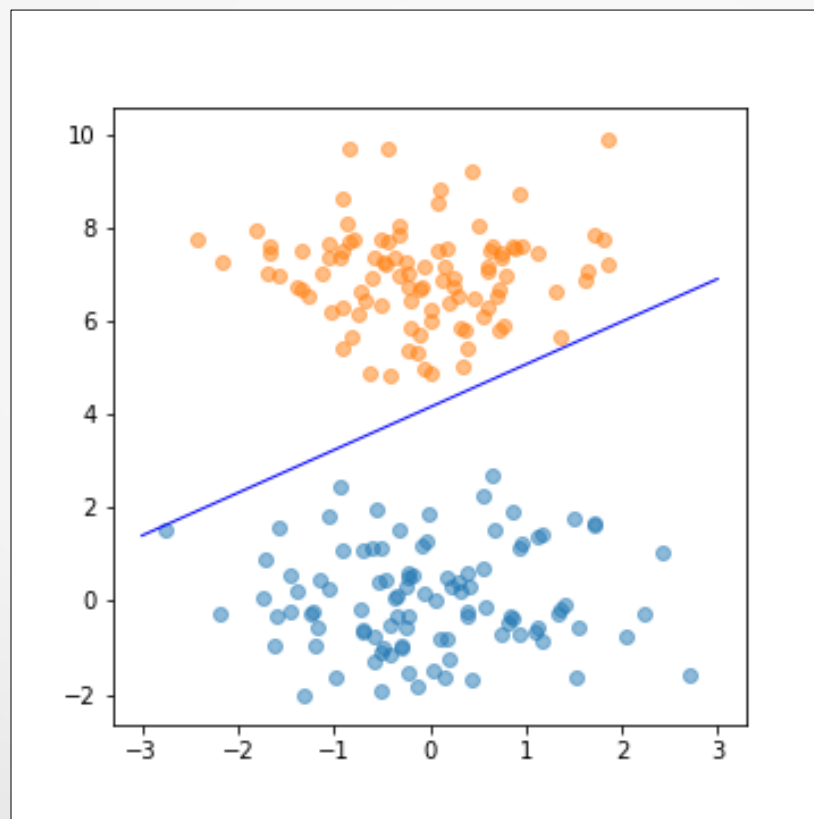
$$f(x, w) = 0$$

Линейные методы: разделяющая поверхность

пример: линейно разделимые данные

разделяющая поверхность - прямая

$$w_1 \cdot x + w_0 = 0$$



Линейные методы: о задаче классификации

метки классов

$$Y = \{-1, 1\}$$

размеченные данные

$$X = (x, y)$$

дискриминантная функция

$$f(x, w)$$

алгоритм классификации

$$a(x, w) = \text{sign}(f(x, w))$$

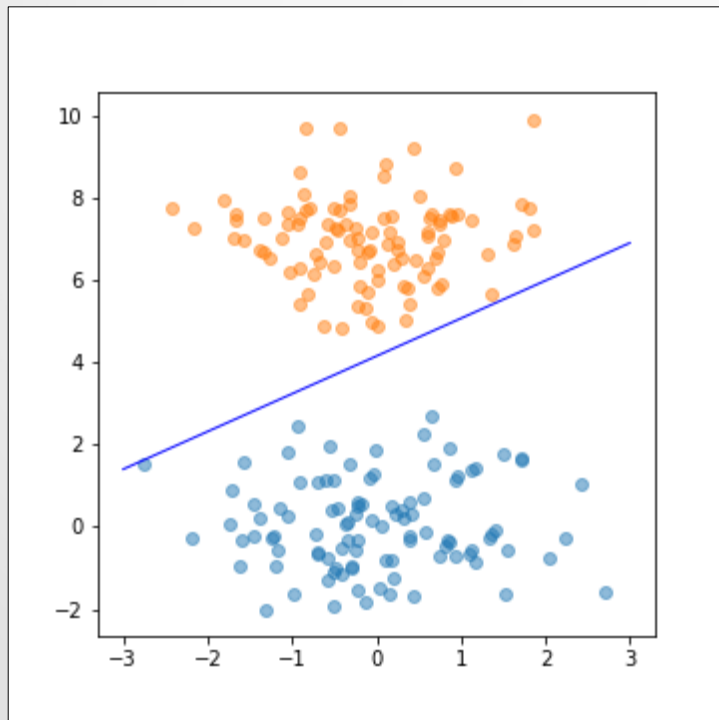
задача:

заданы данные X и вид функции $f(x, w)$
как найти вектор параметров w ?

Линейные методы: отступы

отступ - насколько далеко объект x от разделяющей поверхности

$$M(x, w) = y \cdot f(x, w)$$



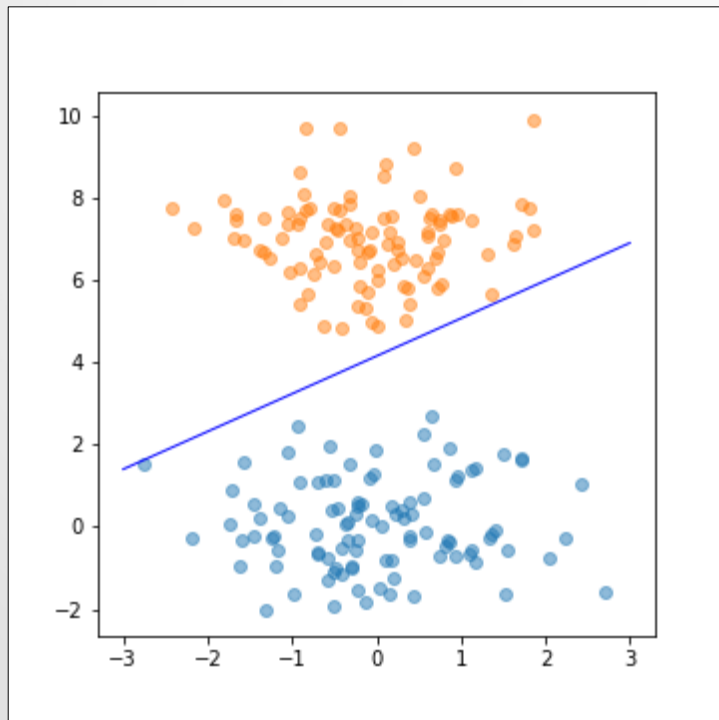
$y \in \{-1, 1\}$ - метка класса

$f(x, w)$ - дискриминантная функция

Линейные методы: отступы

отступ - насколько далеко объект x от разделяющей поверхности

$$M(x, w) = y \cdot f(x, w)$$



$y \in \{-1, 1\}$ - метка класса

$f(x, w)$ - дискриминантная функция

$M(x, w) < 0$ - алгоритм ошибается на x

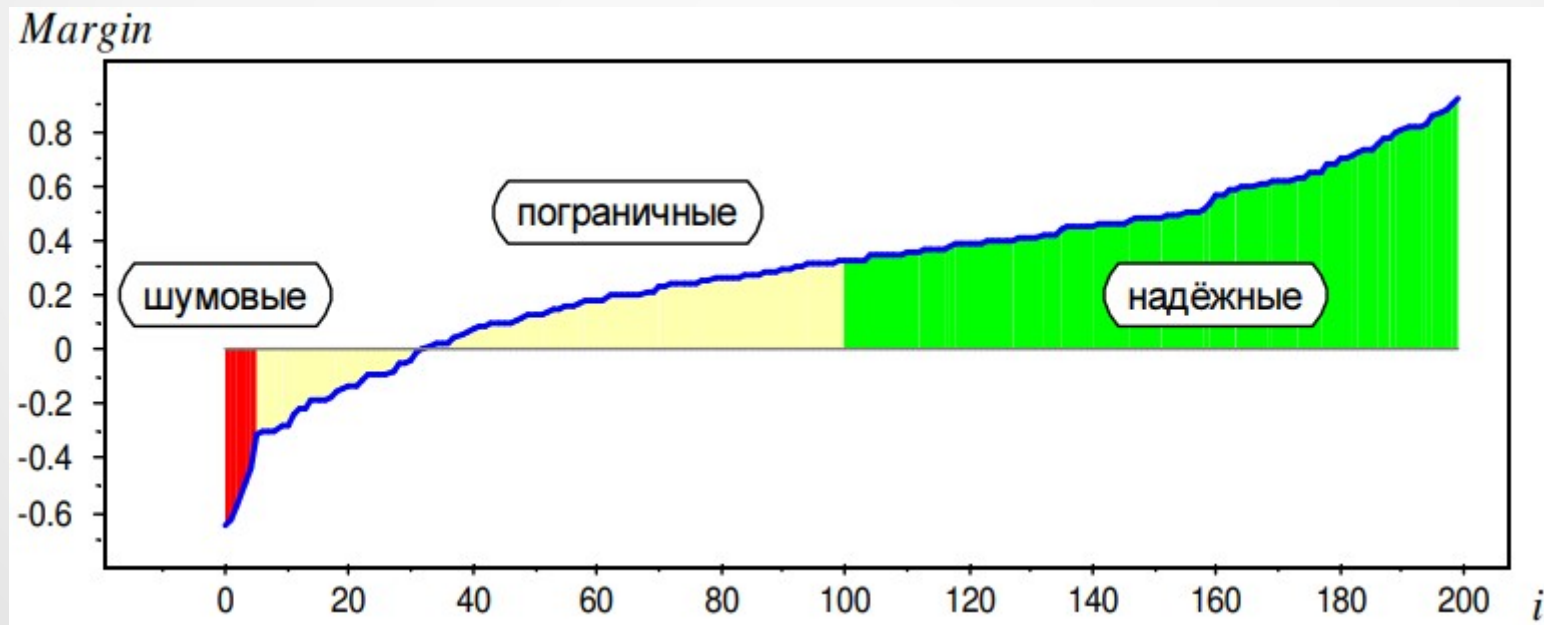
Линейные методы: отступы

отступ - насколько далеко объект от разделяющей поверхности

$$M(x, w) = y \cdot f(x, w)$$

$y \in \{-1, 1\}$ - метка класса

$f(x, w)$ - дискриминантная функция



$M(x, w) < 0$ - алгоритм ошибается на x

Линейные методы: эмпирический риск

функционал эмпирического риска, (число ошибок)

$$Q(x, w) = \sum_x [M(x, w) < 0]$$

$M(x, w) = f(x, w) \cdot y$ - отступ объекта x

$y \in \{-1, 1\}$ - метка класса

$f(x, w)$ - дискриминантная функция

$M(x, w) < 0$ - алгоритм ошибается на x

Линейные методы: функция потерь

функционал эмпирического риска

$$Q(x, w) = \sum_x [M(x, w) < 0]$$

Линейные методы: функция потерь

функционал эмпирического риска

$$Q(x, w) = \sum_x [M(x, w) < 0]$$

[$M < 0$] это пороговая функция,
не учитываем значение отступа M ,
оптимизировать не удобно,
заменим её...

Линейные методы: функция потерь

функционал эмпирического риска

$$Q(x, w) = \sum_x [M(x, w) < 0]$$

[$M < 0$] это пороговая функция,

не учитываем значение отступа M ,

оптимизировать не удобно,

заменим её...

построим аппроксимацию Q

введём **функцию потерь $L(M)$**

(невозрастающая, неотрицательная)

$$\tilde{Q}(x, w) = \sum_x L(M(x, w)) \rightarrow \min$$

$$Q(x, w) \leq \tilde{Q}(x, w)$$

Линейные методы

функционал эмпирического риска

$$Q(x, w) = \sum_x [M(x, w) < 0]$$

$[M < 0]$ это пороговая функция, оптимизировать не удобно, заменим её...

варианты для замены $[M < 0]$

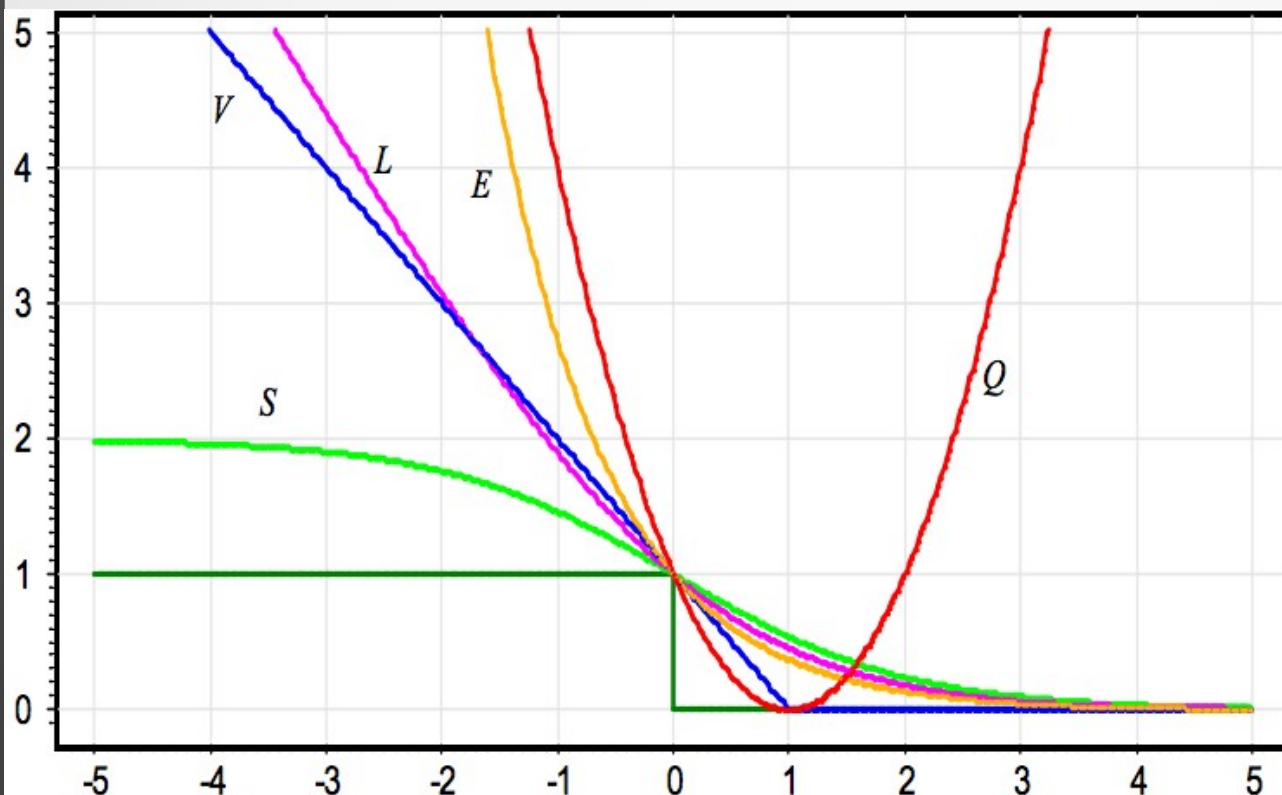
$$L(M) = \log_2 \left(1 + \frac{1}{\exp(M)} \right) \quad \text{логарифмическая}$$

$$V(M) = (1 - M)_+ \quad \text{кусочно-линейная}$$

$$Q(M) = (1 - M)^2 \quad \text{квадратичная}$$

$$E(M) = \frac{1}{\exp(M)} \quad \text{экспоненциальная}$$

$$S(M) = \frac{1}{2 \cdot (1 + \exp(M))} \quad \text{сигмоид}$$



Линейные методы: линейный классификатор

рассмотрим линейный классификатор,

дискриминантная функция $f(x, w)$ это гиперплоскость

$$f(x, w) = \sum_{i=1}^n x_i \cdot w_i - w_0 \quad - \text{дискриминантная функция}$$

$$a(x, w) = \text{sign}(f(x, w)) = \text{sign}\left(\sum_{i=1}^n x_i \cdot w_i - w_0\right) = \text{sign}(\langle x, w \rangle)$$

$$M(x, w) = \langle x, w \rangle \cdot y \quad - \text{отступ на объекте } \mathbf{x} \text{ класса } \mathbf{y}$$

$$\text{sign}(x) = \begin{cases} 1, & x > 0 \\ 0, & x = 0 \\ -1, & x < 0 \end{cases}$$

Линейные методы

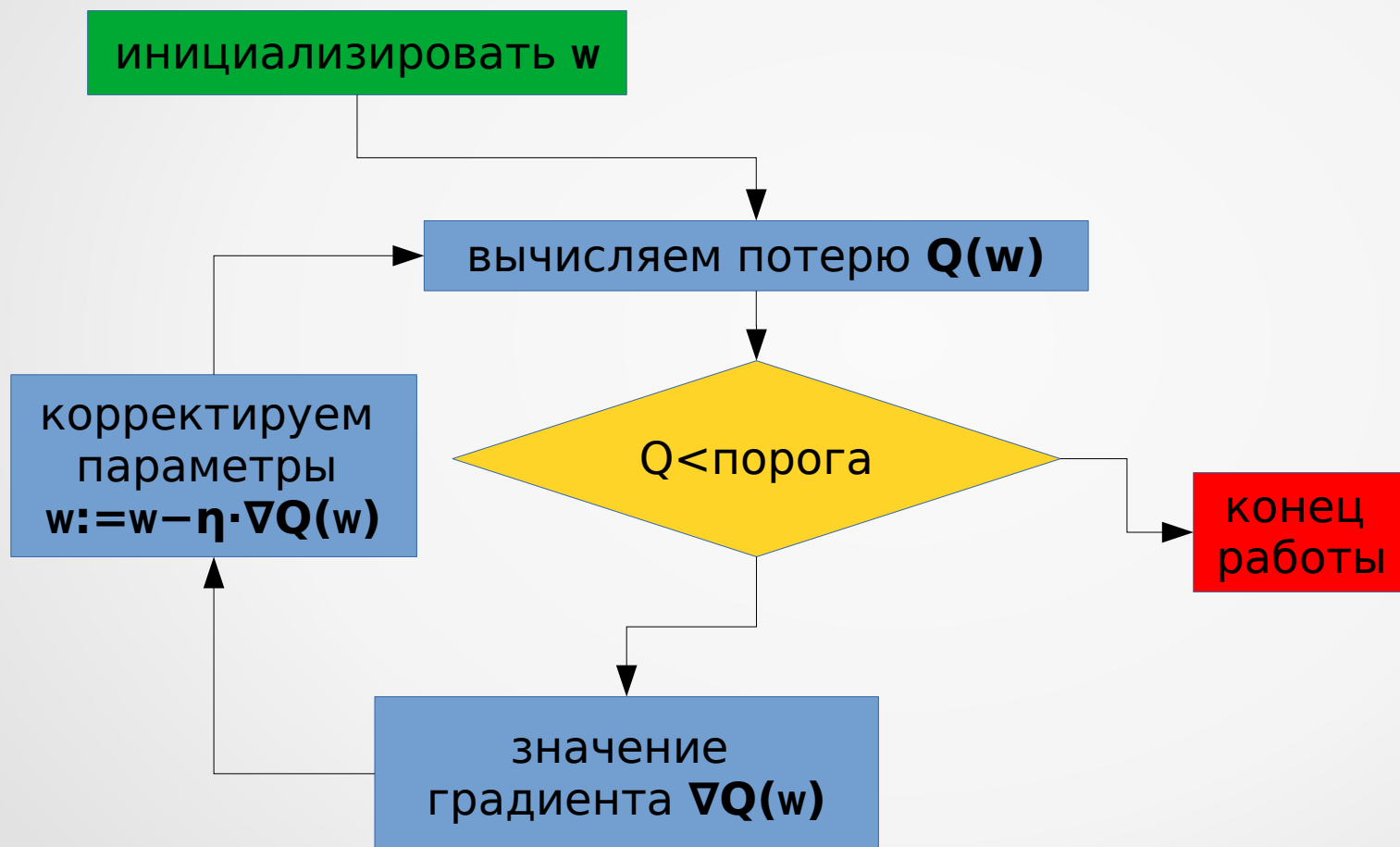
обучение классификатора как задача оптимизации

$$Q(w; X) = \sum_{x \in X} L(\langle x, w \rangle \cdot y) \rightarrow \min_w$$

можно использовать градиентные методы

$$\nabla Q(w) = \left(\frac{\partial Q(w)}{\partial w_j} \right)_{j=0}^n \text{ - вектор градиента ф-ции } Q$$

Линейные методы: градиентный спуск (GD)



Линейные методы: стохастический градиентный спуск (SGD)

инициализировать w

вычисляем суммарную
потерю $Q(w)$ на X

$Q < \text{порога}$

конец
работы

выбираем
случайный x_j

вычисляем значение
градиента $\nabla Q(w, x_j)$

корректируем
параметры
 $w := w - \eta \cdot \nabla Q(w, x_j)$

вычисляем
потерю для объекта x_j
 $Q_j = Q(w, x_j)$

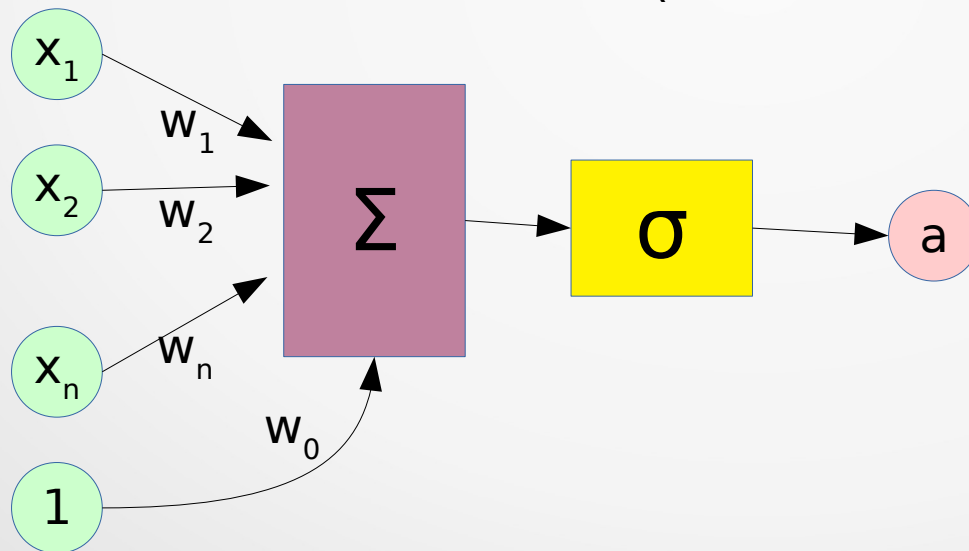
корректируем
суммарную потерю
 $Q := \lambda Q_j + (1 - \lambda)Q$

Линейные методы: линейный классификатор

линейная модель МакКаллока-Питтса (1943) (формальный нейрон)

$$a(x, w) = \sigma \left(\sum_{i=1}^n x_i \cdot w_i - w_0 \right) = \sigma(\langle x, w \rangle)$$

σ - функция активации нейрона
(можно использовать **sign**)



Линейные методы

правило обучения Хебба (1949)

задача классификации

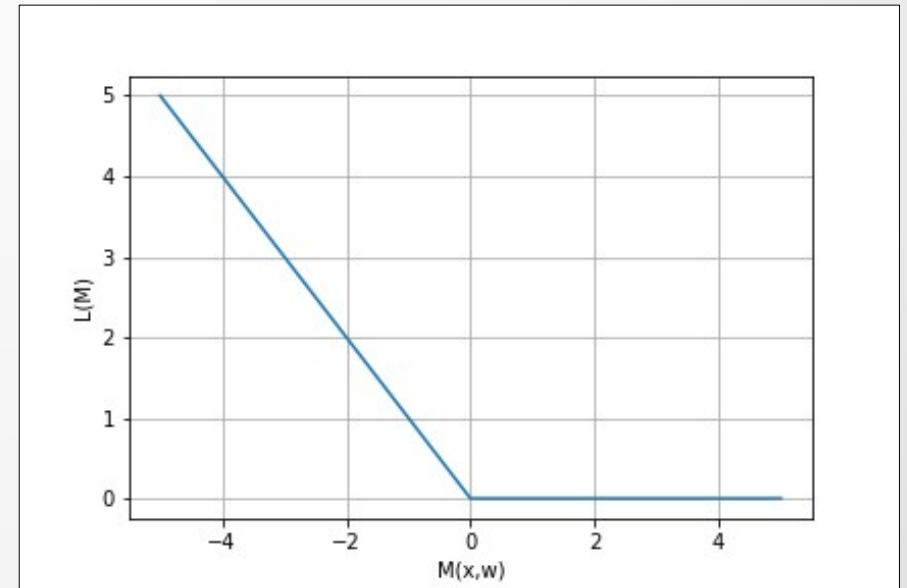
$$x \in \mathbb{R}^n \quad y \in \{-1, 1\} \quad a(x, w) = \text{sign}(\langle x, w \rangle)$$

Линейные методы

правило обучения Хебба (1949)

задача классификации

$$x \in \mathbb{R}^n \quad y \in \{-1, 1\} \quad a(x, w) = \text{sign}(\langle x, w \rangle) \quad L(a, y) = (-\langle x, w \rangle \cdot y)_+$$



Линейные методы

правило обучения Хебба (1949)

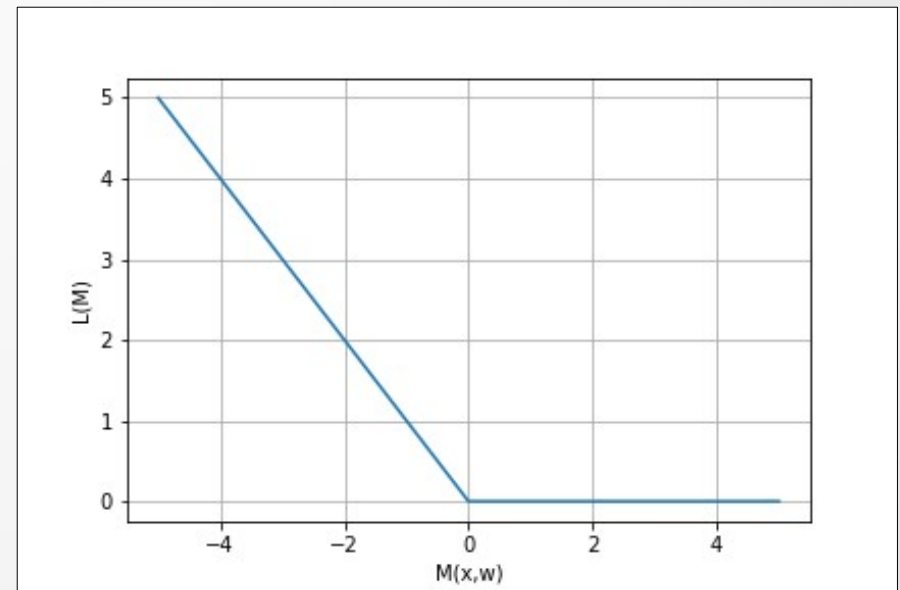
задача классификации

$$x \in \mathbb{R}^n \quad y \in \{-1, 1\} \quad a(x, w) = \text{sign}(\langle x, w \rangle) \quad L(a, y) = (-\langle x, w \rangle \cdot y)_+$$

градиентный шаг

$$[\langle x, w \rangle \cdot y < 0] \Rightarrow w := w + \eta \cdot y \cdot x$$

параметры корректируем
только в случае ошибки



Линейные методы

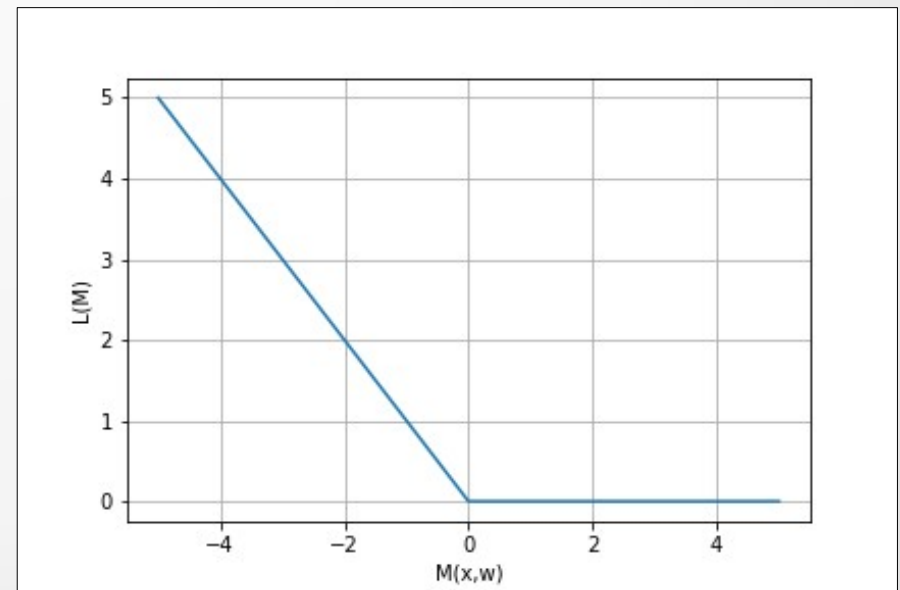
правило обучения Розенблатта (1957)

задача классификации

$$x \in \{0,1\}^n \quad y \in \{0,1\} \quad a(x, w) = \text{sign}(\langle x, w \rangle) \quad L(a, y) = (-\langle x, w \rangle \cdot y)_+$$

градиентный шаг

$$w := w - \eta \cdot (a(x, w) - y) \cdot x$$



Линейные методы

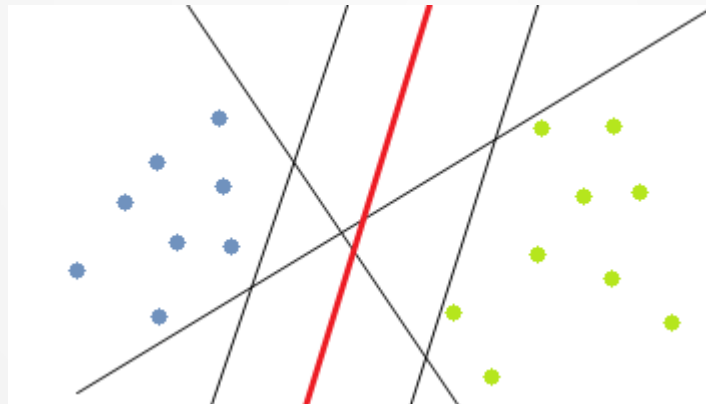
«зоопарк» методов

- вид разделяющей поверхности $f(\mathbf{x}, \mathbf{w})$
(линейная, нелинейная)
- вид функции потерь $L(\mathbf{M})$
- вид метода оптимизации $Q(\mathbf{w}) \rightarrow \min$

Линейные методы: SVM

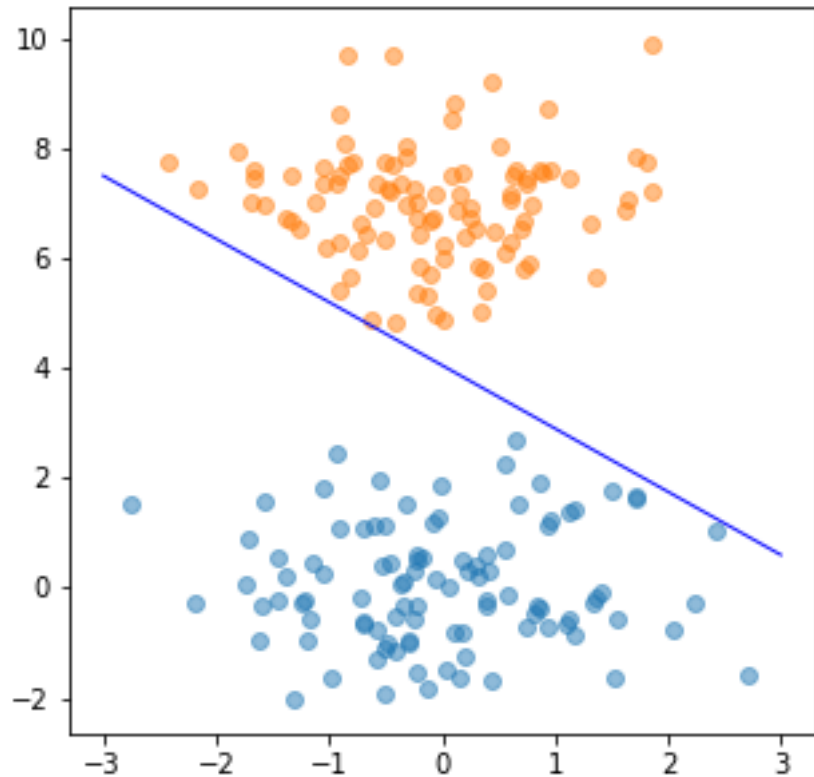
Метод опорных векторов (SVM, support vector machine)

В.Н.Вапник, А.Я.Червоненкис, (1963)



Линейные методы: SVM

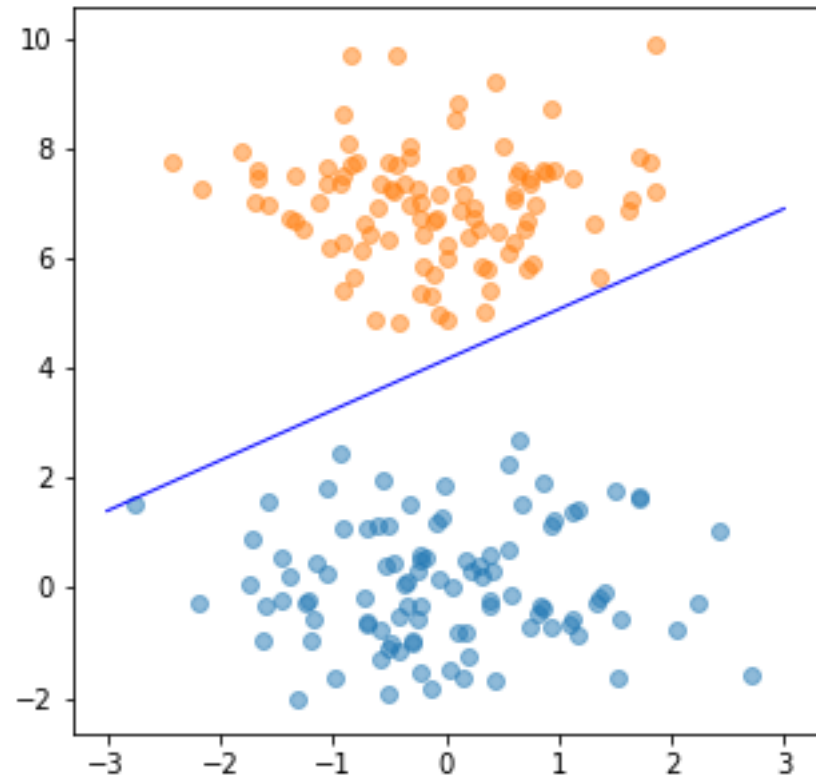
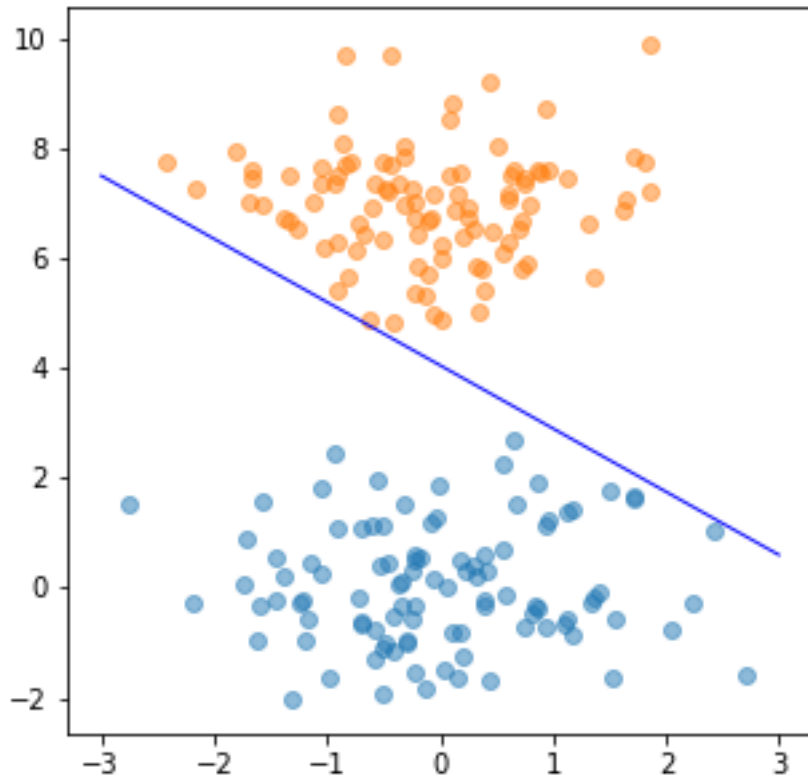
рассмотрим линейно разделимый набор



Линейные методы: SVM

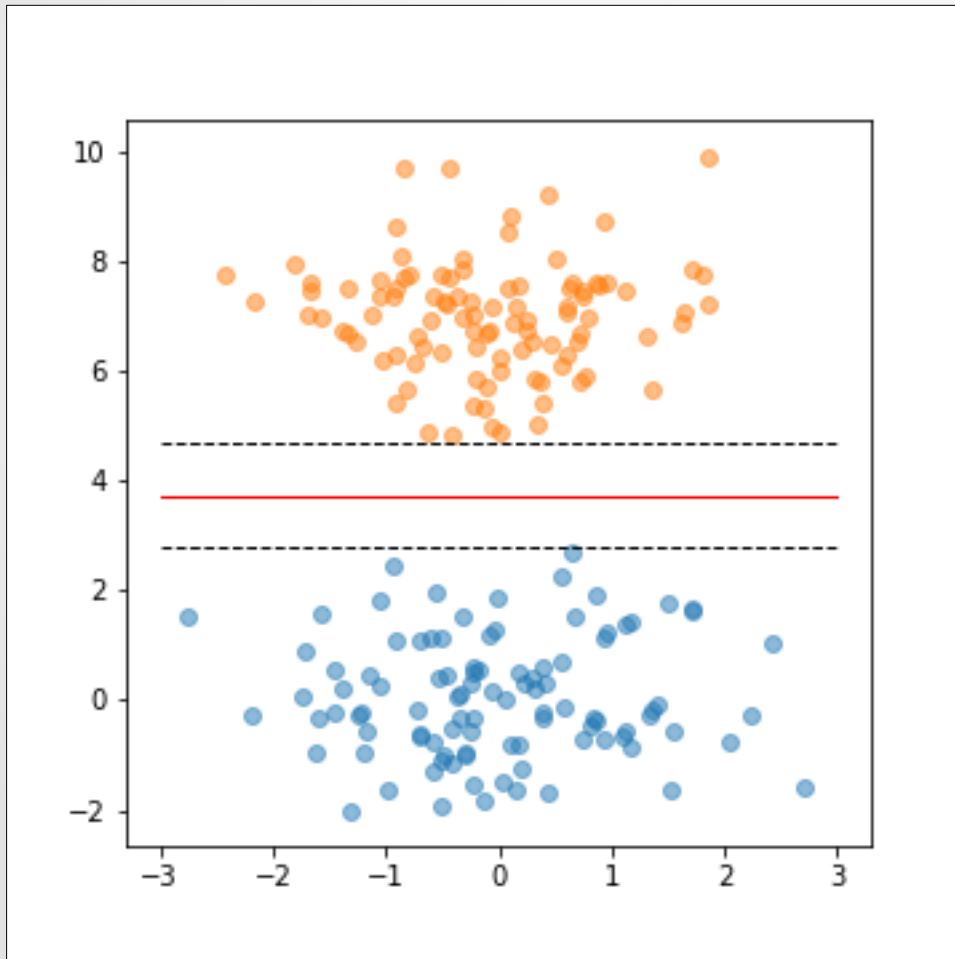
рассмотрим линейно разделимый набор

много разделяющих гиперплоскостей



Линейные методы: SVM

разделительная полоса



цель: увеличить отступы,
получить полосу максимальной ширины

Линейные методы: SVM

$$a(x) = \text{sign}\left(\sum_i \lambda_i y_i \langle x_i, x \rangle - w_0\right)$$

опорным назовём объект x_i , для которого $\lambda_i \neq 0$

для нахождения опорных объектов
применяется алгоритм SMO (sequential minimal optimization)

метод обучения SVM как задача выпуклой квадратичной
оптимизации имеет единственное решение

Линейные методы: SVM

нелинейное обобщение - kernel trick

вместо скалярного произведения

будем использовать функцию-ядро

$$a(x) = \text{sign} \left(\sum_i \lambda_i y_i K(x_i, x) - w_0 \right)$$

функция K - ядро

если для него существует отображение,
удовлетворяющее условиям скалярного произведения

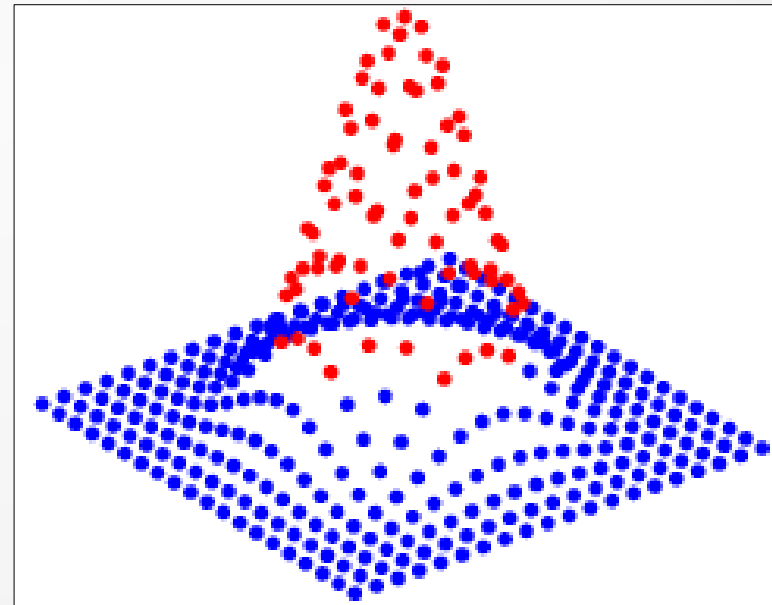
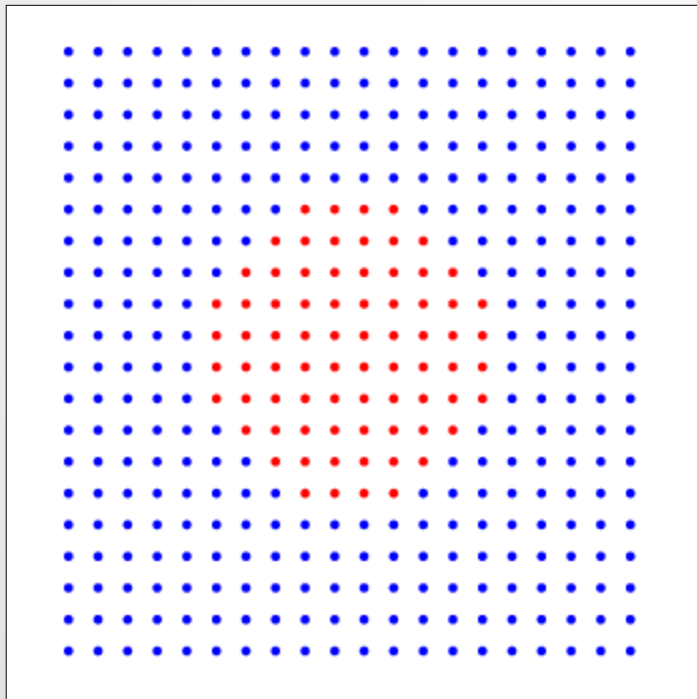
$$\exists \psi: K(x, x') = \langle \psi(x), \psi(x') \rangle$$

функция K симметрична и неотрицательно определена

Линейные методы: SVM

kernel trick

с помощью ядра отображаем данные
в пространство большей размерности
линейно неразделимая задача
превращается в линейно разделимую



Линейные методы: SVM

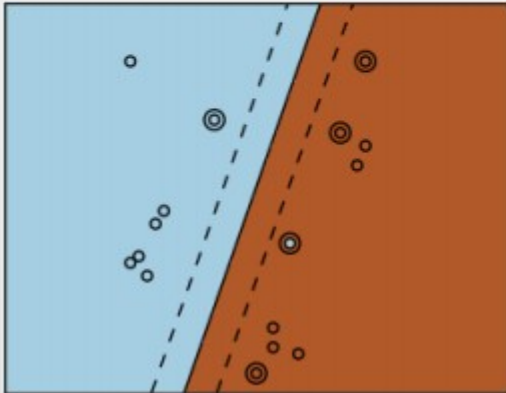
kernel trick

с помощью ядра отображаем данные
в пространство большей размерности
линейно неразделимая задача
превращается в линейно разделимую

Примеры с различными ядрами $K(x, x')$

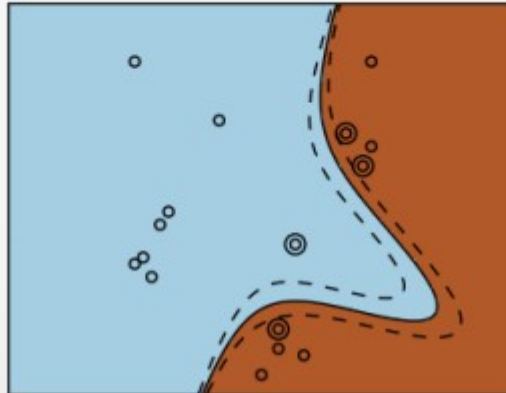
линейное

$$\langle x, x' \rangle$$



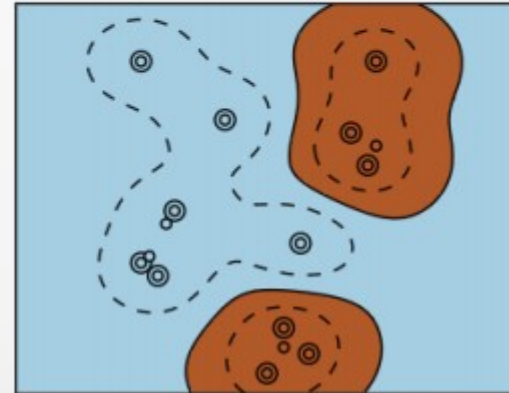
полиномиальное

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



гауссовское (RBF)

$$\exp(-\gamma \|x - x'\|^2)$$



Линейные методы: итог

- линейные методы строят разделяющие поверхности в пространстве признаков
- использования нелинейных поверхностей позволяет разделять линейно неразделимые наборы
- аппроксимация пороговой ф-ции потерь позволяет использовать градиентные методы оптимизации
- метод стохастического градиента SGD подходит для обучения на больших данных
- для обучения SVM применяется алгоритм SMO (sequential minimal optimization)
- применение ядер позволяет SVM разделять линейно неразделимые наборы, общих подходов для выбора ядер нет

Линейные методы: литература

git clone https://github.com/mechanoid5/ml_lectorium.git

- Борисов Е.С. Классификатор на основе машины опорных векторов.
<http://mechanoid.kiev.ua/ml-svm.html>
- К.В. Воронцов Линейные методы классификации: метод стохастического градиента.
- К.В. Воронцов Линейные методы классификации: метод опорных векторов.

Линейные методы



Вопросы ?

Линейные методы: практика



источники данных для экспериментов

sklearn.datasets
UCI Repository
kaggle



практика

- разделить данные на train/test (sklearn.train_test_split)
- посчитать метрики качества (confusion matrix, precision, recall, ROC/AUC)
- классифицировать данные с использованием sklearn.SVM