Лекция 5
Линейные модели классификации. Часть 2.

Кантонистова Е.О.

ВШЭ, 2018

ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

Логистическая регрессия – линейный классификатор, корректно предсказывающий вероятности классов.

ВЕРОЯТНОСТНАЯ ПОСТАНОВКА ЗАДАЧИ

Предположение: В каждой точке x пространства объектов задана вероятность $p(y = +1|x)$

Объекты с одинаковым признаковым описанием могут иметь разные значения целевой переменной.

ВЕРОЯТНОСТНАЯ ПОСТАНОВКА ЗАДАЧИ

Предположение: В каждой точке x пространства объектов задана вероятность $p(y = +1|x)$

Объекты с одинаковым признаковым описанием могут иметь разные значения целевой переменной.

Цель: построить алгоритм $b(x)$, в каждой точке x предсказывающий $p(y = +1|x)$.

ВЕРОЯТНОСТНАЯ ПОСТАНОВКА ЗАДАЧИ

- Пусть объект x встречается в выборке n раз с ответами $\{y_1, \dots, y_n\}$. Хотим, чтобы алгоритм выдавал вероятность классов:

$$b_* = \underset{b \in \mathbb{R}}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} L(y_i, b) \approx p(y = +1|x)$$

По ЗБЧ при $n \to \infty$ получаем

$$b_* = \underset{b \in \mathbb{R}}{\operatorname{argmin}} E[L(y, b)|x] = p(y = +1|x)$$

ФУНКЦИИ ПОТЕРЬ

Подходят:

- **Квадратичная**

$$L(y,z) = (y-z)^2$$

- **Логистическая**

$$L(y,z) = [y=+1]\cdot\log\big(b(x,w)\big) + [y=-1]\cdot\log(1-b(x,w))$$

Не подходят:

- **Модуль**

$$L(y,z) = |\mathrm{y}-\mathrm{z}|$$

ПРАВДОПОДОБИЕ И LOG-LOSS

- Вероятности, которые выдает алгоритм $b(x)$, должны согласовываться с выборкой
- Вероятность того, что в выборке встретится объект x с классом y:

$$b(x)^{[y=+1]} \cdot \big(1 - b(x)\big)^{[y=-1]}$$

ПРАВДОПОДОБИЕ И LOG-LOSS

- Вероятности, которые выдает алгоритм $b(x)$, должны согласовываться с выборкой
- Вероятность того, что в выборке встретится объект x с классом y:

$$b(x)^{[y=+1]} \cdot \big(1 - b(x)\big)^{[y=-1]}$$

Правдоподобие выборки:

$$(b, X) = \prod_{i=1}^{l} b(x_i)^{[y_i=+1]} \cdot \big(1 - b(x_i)\big)^{[y_i=-1]}$$

ФУНКЦИЯ ПОТЕРЬ ДЛЯ ОБУЧЕНИЯ

- Можно максимизировать правдоподобие

$$(b, X) = \prod_{i=1}^{l} b(x_i)^{[y_i=+1]} \cdot \big(1 - b(x_i)\big)^{[y_i=-1]} \to \max_b$$

- Или, что эквивалентно (**логарифмическая, log-loss**):

$$-\sum_{i=1}^{l} ([y_i = +1] \log b(x_i) + [y_i = -1] \log(1 - b(x_i))) \to \min_b$$

ФУНКЦИЯ ПОТЕРЬ ДЛЯ ОБУЧЕНИЯ

- Можно максимизировать правдоподобие

$$(b, X) = \prod_{i=1}^{l} b(x_i)^{[y_i=+1]} \cdot \big(1 - b(x_i)\big)^{[y_i=-1]} \to \max_b$$

- Или, что эквивалентно (**логарифмическая, log-loss**):

$$-\sum_{i=1}^{l} ([y_i = +1] \log b(x_i) + [y_i = -1] \log(1 - b(x_i))) \to \min_b$$

Утверждение. Логарифмическая функция потерь корректно предсказывает вероятности.

ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

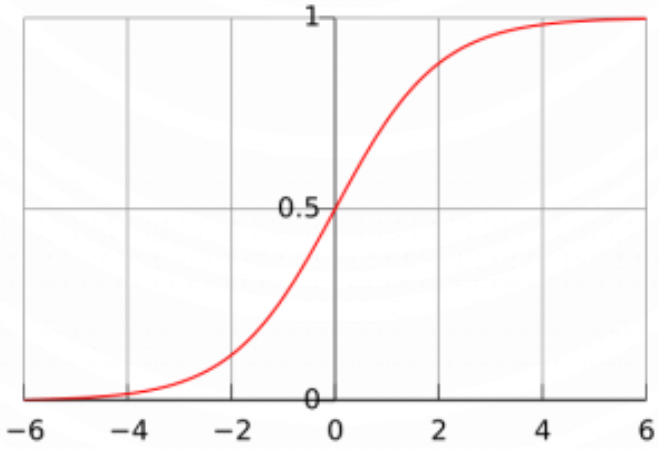
- Хотим, чтобы алгоритм $b(x)$ возвращал числа из отрезка $[0, 1]$.

ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

- Хотим, чтобы алгоритм $b(x)$ возвращал числа из отрезка $[0, 1]$.
- Можно взять $b(x) = \sigma(w^T x)$, где σ – любая монотонно неубывающая функция с областью значений $[0, 1]$.

ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

- Хотим, чтобы алгоритм $b(x)$ возвращал числа из отрезка $[0, 1]$.
- Можно взять $b(x) = \sigma(w^T x)$, где σ – любая монотонно неубывающая функция с областью значений $[0, 1]$.
- Возьмем ***сигмоиду***: $\sigma(z) = \frac{1}{1+e^{-z}}$



ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

- $p(y = +1|x) = \frac{1}{1+e^{-w^T x}}$, следовательно,
- $(w, x) = w^T x = \log \frac{p(y=+1|x)}{p(y=-1|x)}$ - логарифм отношения вероятностей классов.

Утверждение. Логарифмическая функция потерь может быть записана в виде

$$\boldsymbol{L(b, X) = \sum_{i=1}^{l} \log(1 + e^{-y_i(w,x)})}$$

МЕТОД ОПОРНЫХ ВЕКТОРОВ

ЛИНЕЙНЫЙ КЛАССИФИКАТОР

- $a(x) = sign\big((w, x) + w_0\big)$

Ошибка линейного классификатора:

- $Q(a, X) = \frac{1}{l}\sum_{i=1}^{l}[a(x_i) \neq y_i] =$

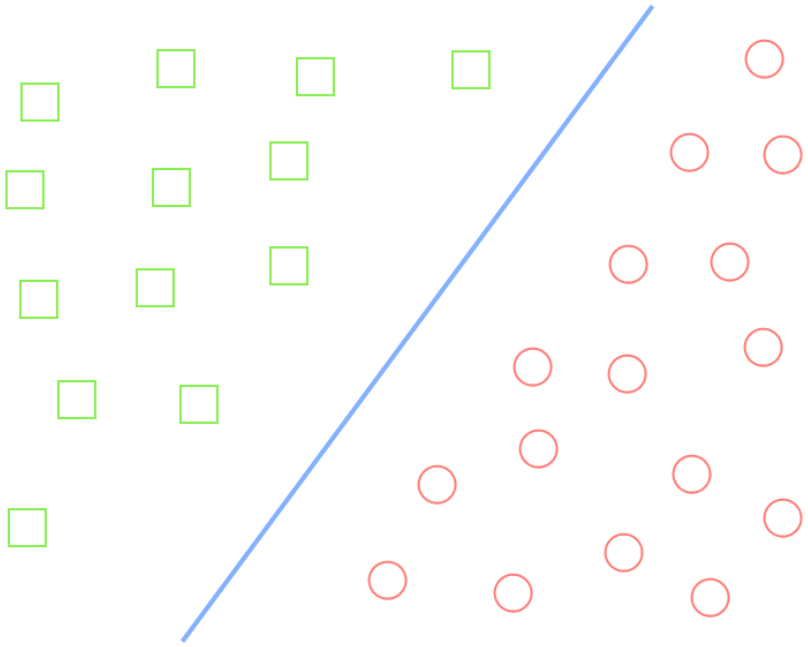
$$= \frac{1}{l}\sum_{i=1}^{l}\big[sign\big((w, x_i) + w_0\big) \neq y_i\big] =$$

$$= \frac{1}{l}\sum_{i=1}^{l}[\boldsymbol{y_i \cdot ((w, x_i) + w_0)} < 0] \to \min_{w, w_0}$$

$M_i = y_i((w, x_i) + w_0)$ – отступ на объекте

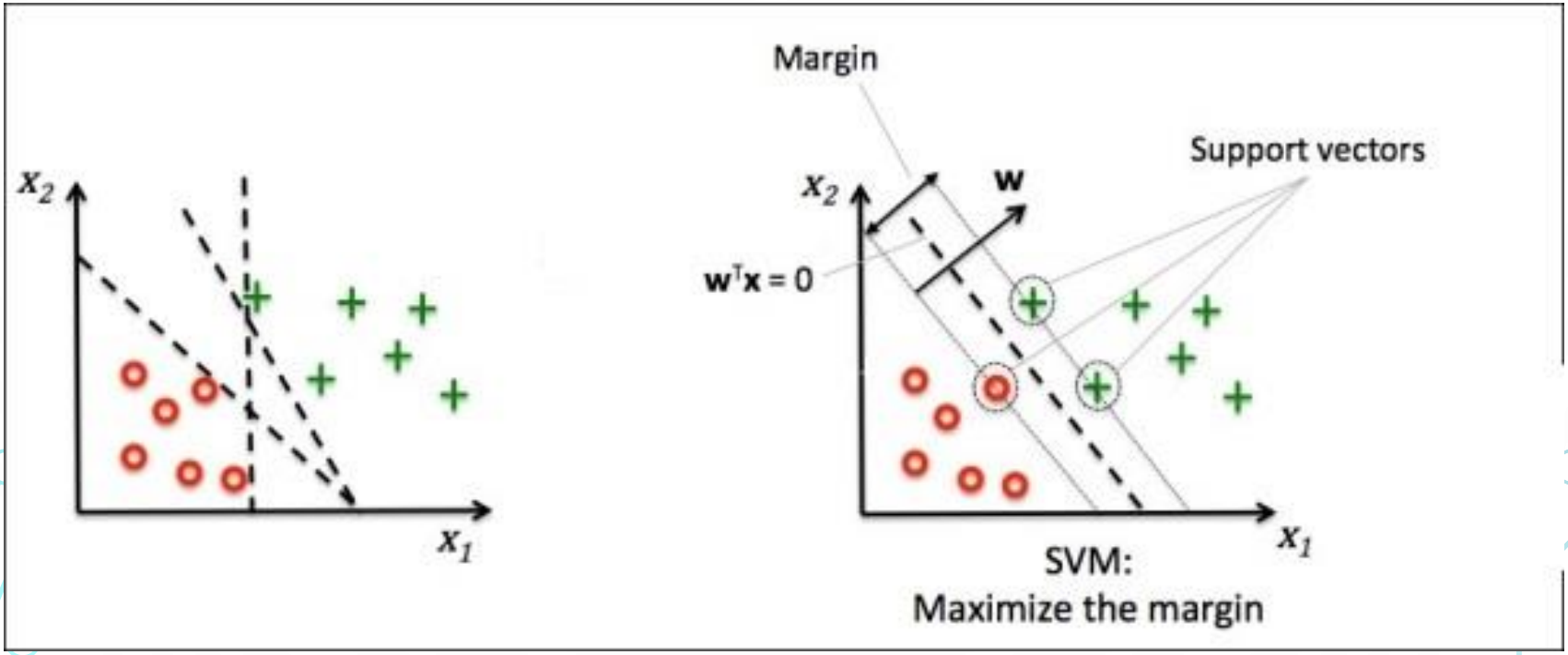
ЛИНЕЙНО РАЗДЕЛИМАЯ ВЫБОРКА

Выборка ***линейно разделима***, если существует такой вектор параметров w^*, что соответствующий классификатор $a(x)$ не допускает ошибок на этой выборке.



МЕТОД ОПОРНЫХ ВЕКТОРОВ: РАЗДЕЛИМЫЙ СЛУЧАЙ

Цель метода опорных векторов (Support Vector Machine) – максимизировать ширину разделяющей полосы.



МЕТОД ОПОРНЫХ ВЕКТОРОВ: РАЗДЕЛИМЫЙ СЛУЧАЙ

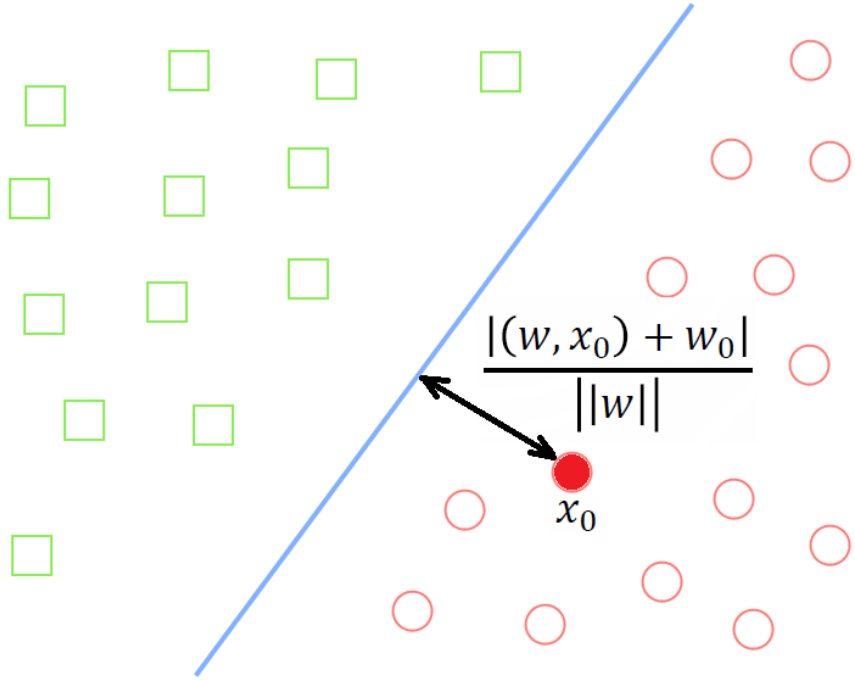
- $a(x) = sign\big((w, x) + w_0\big)$
- Нормируем параметры w и w_0 так, что

$$\min_{x \in X} |(w, x) + w_0| = 1$$

Расстояние от точки x_0 до разделяющей гиперплоскости, задаваемой

классификатором:

$$\rho(x_0, a) = \frac{|(w, x_0) + w_0|}{||w||}$$



МЕТОД ОПОРНЫХ ВЕКТОРОВ: РАЗДЕЛИМЫЙ СЛУЧАЙ

- Нормируем параметры w и w_0 так, что

$$\min_{x \in X} |(w, x) + w_0| = 1$$

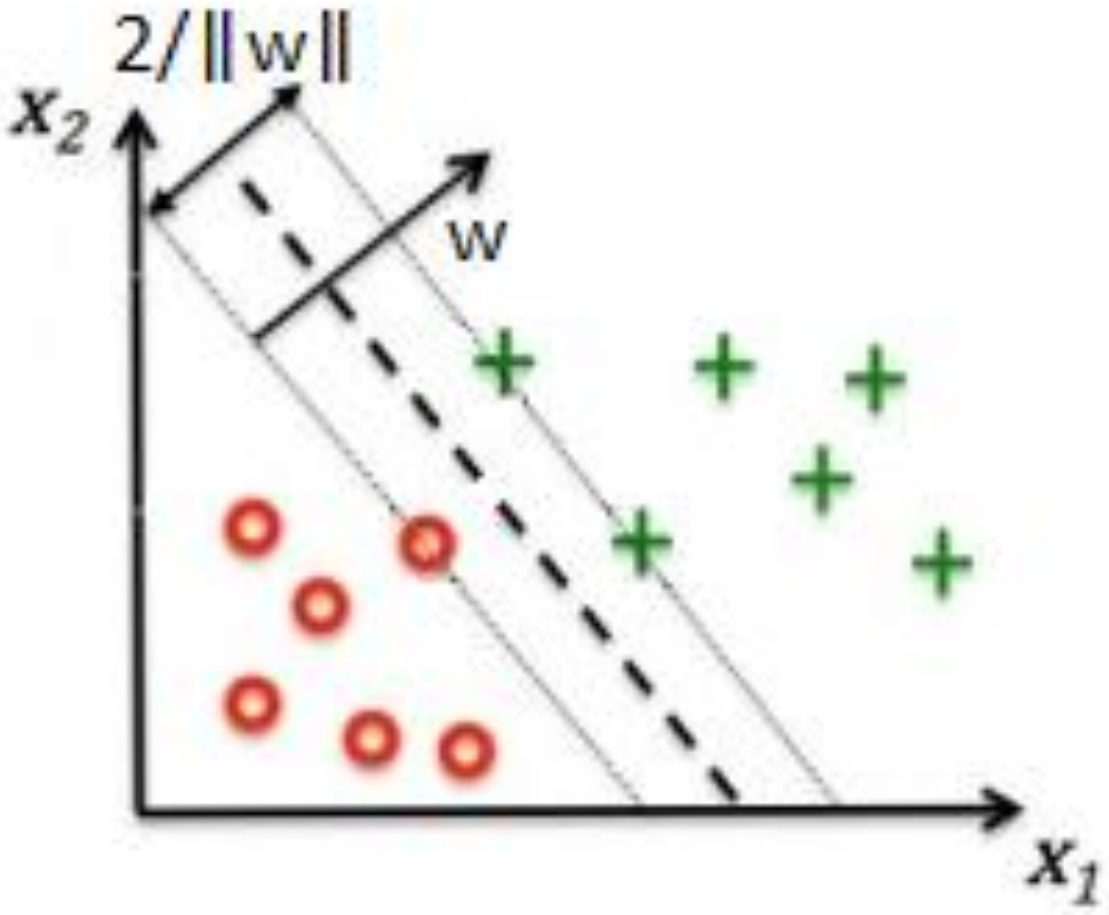
Тогда расстояние от точки x_0 до разделяющей гиперплоскости, задаваемой классификатором:

$$\rho(x_0, a) = \frac{|(w, x_0) + w_0|}{||w||}$$

- Расстояние до ближайшего объекта $x \in X$:

$$\min_{x \in X} \frac{|(w, x) + w_0|}{||w||} = \frac{1}{||w||} \min_{x \in X} |(w, x) + w_0| = \frac{\mathbf{1}}{\mathbf{||w||}}$$

РАЗДЕЛЯЮЩАЯ ПОЛОСА



ОПТИМИЗАЦИОННАЯ ЗАДАЧА SVM ДЛЯ РАЗДЕЛИМОЙ ВЫБОРКИ

$$\begin{cases} \frac{1}{2}\left||w|\right|^2 \to \min\limits_{w} \\ y_i\big((w, x_i) + w_0\big) \geq 1, i = 1, \dots, l \end{cases}$$

Утверждение. Данная оптимизационная задача имеет единственное решение.

ЛИНЕЙНО НЕРАЗДЕЛИМАЯ ВЫБОРКА

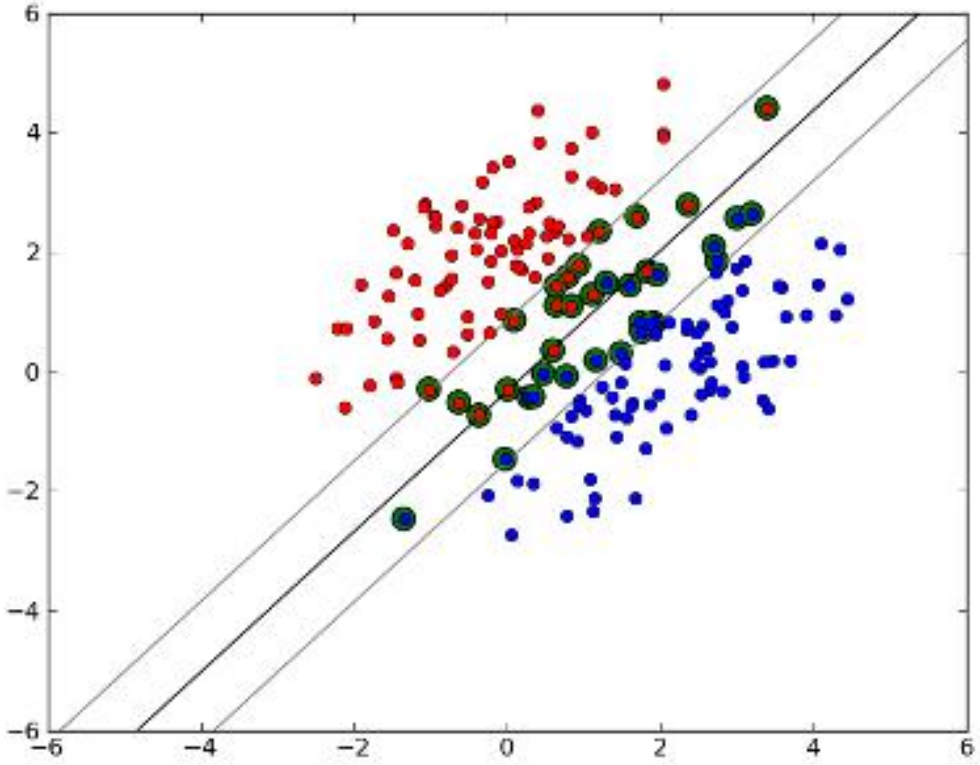
- Существует хотя бы один объект $x \in X$, что

$$y_i\big((w, x_i) + w_0\big) < 1$$

ЛИНЕЙНО НЕРАЗДЕЛИМАЯ ВЫБОРКА

- Существует хотя бы один объект $x \in X$, что

$$y_i\big((w, x_i) + w_0\big) < 1$$



ЛИНЕЙНО НЕРАЗДЕЛИМАЯ ВЫБОРКА

- Существует хотя бы один объект $x \in X$, что

$$y_i\big((w, x_i) + w_0\big) < 1$$

Смягчим ограничения, введя штрафы $\xi_i \geq 0$:

$$y_i\big((w, x_i) + w_0\big) \geq 1 - \xi_i, i = 1, \dots, l$$

МЕТОД ОПОРНЫХ ВЕКТОРОВ: НЕРАЗДЕЛИМЫЙ СЛУЧАЙ

Хотим:

- Минимизировать штрафы $\sum_{i=1}^{l} \xi_i$
- Максимизировать отступ $\frac{1}{||w||}$

МЕТОД ОПОРНЫХ ВЕКТОРОВ: НЕРАЗДЕЛИМЫЙ СЛУЧАЙ

Хотим:

- Минимизировать штрафы $\sum_{i=1}^{l} \xi_i$
- Максимизировать отступ $\frac{1}{||w||}$

Задача оптимизации:

$$\begin{cases} \frac{1}{2}||w||^2 + C\sum_{i=1}^{l} \xi_i \to \min\limits_{w, w_0, \xi_i} \\ y_i\big((w, x_i) + w_0\big) \geq 1 - \xi_i, i = 1, \dots, l \\ \xi_i \geq 0, i = 1, \dots, l \end{cases}$$

МЕТОД ОПОРНЫХ ВЕКТОРОВ: НЕРАЗДЕЛИМЫЙ СЛУЧАЙ

Утверждение. Задача

$$\begin{cases} \frac{1}{2}||w||^2 + C\sum_{i=1}^{l}\xi_i \to \min\limits_{w,w_0,\xi_i} \\ y_i\big((w,x_i)+w_0\big) \geq 1-\xi_i, i=1,\dots,l \\ \xi_i \geq 0, i=1,\dots,l \end{cases}$$

Является выпуклой и имеет единственное решение.

СВЕДЕНИЕ К БЕЗУСЛОВНОЙ ЗАДАЧЕ

$$\begin{cases} \frac{1}{2}||w||^2 + C\sum_{i=1}^{l}\xi_i \to \min\limits_{w,w_0,\xi_i} (1) \\ y_i\big((w,x_i)+w_0\big) \geq 1-\xi_i, i=1,\dots,l\ (2) \\ \xi_i \geq 0, i=1,\dots,l\ (3) \end{cases}$$

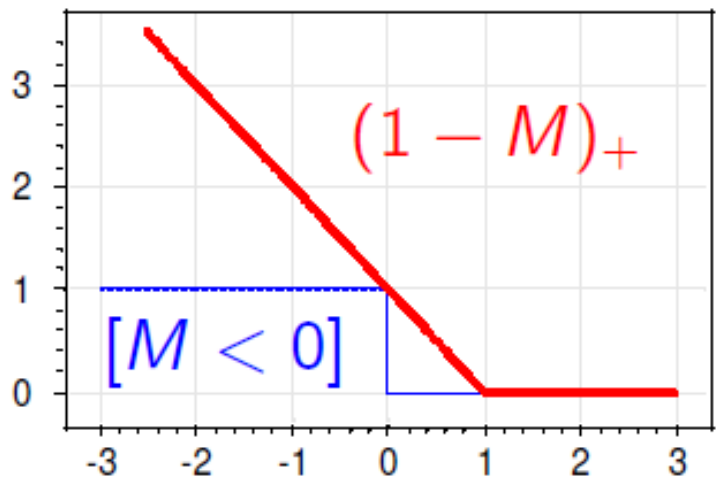
Задача метода опорных векторов (задача условной оптимизации) эквивалентна следующей задаче безусловной оптимизации:

$$\boldsymbol{\frac{1}{2}||w||^2 + C\sum_{i=1}^{l}\max\Big(0, 1-y_i\big((w,x_i)+w_0\big)\Big) \to \min\limits_{w,w_0}}$$

МЕТОД ОПОРНЫХ ВЕКТОРОВ: ЗАДАЧА ОПТИМИЗАЦИИ

- На задачу оптимизации SVM можно смотреть, как на *оптимизацию функции потерь* $L(M) = max(0,1 - M) = (1 - M)_+$ *с регуляризацией*:

$$Q(a, X) = \sum_{i=1}^{l} \big(1 - M_i(w, w_0)\big)_+ + \frac{1}{2C}||w||^2 \to \min_{w, w_0}$$

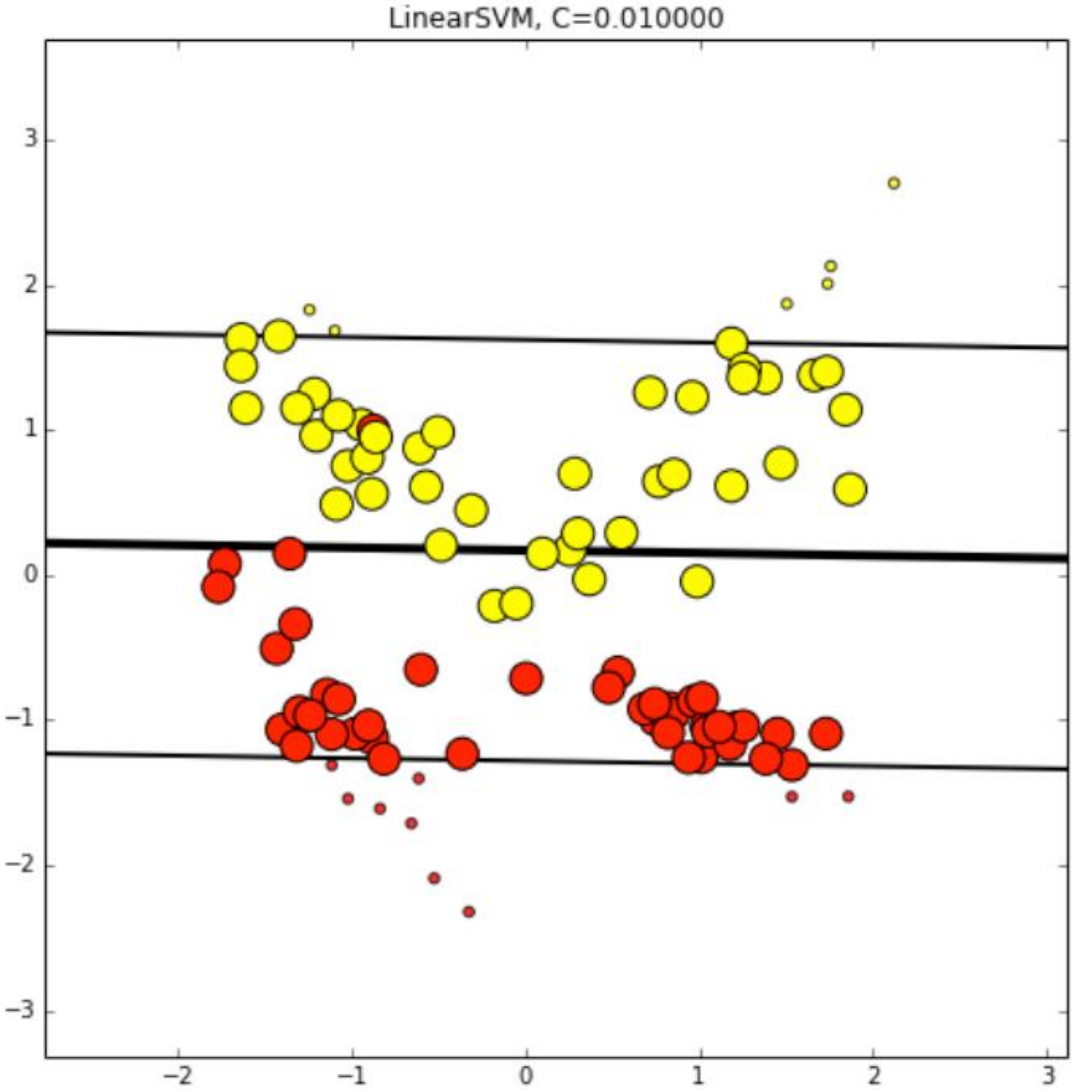


ЗНАЧЕНИЕ КОНСТАНТЫ C

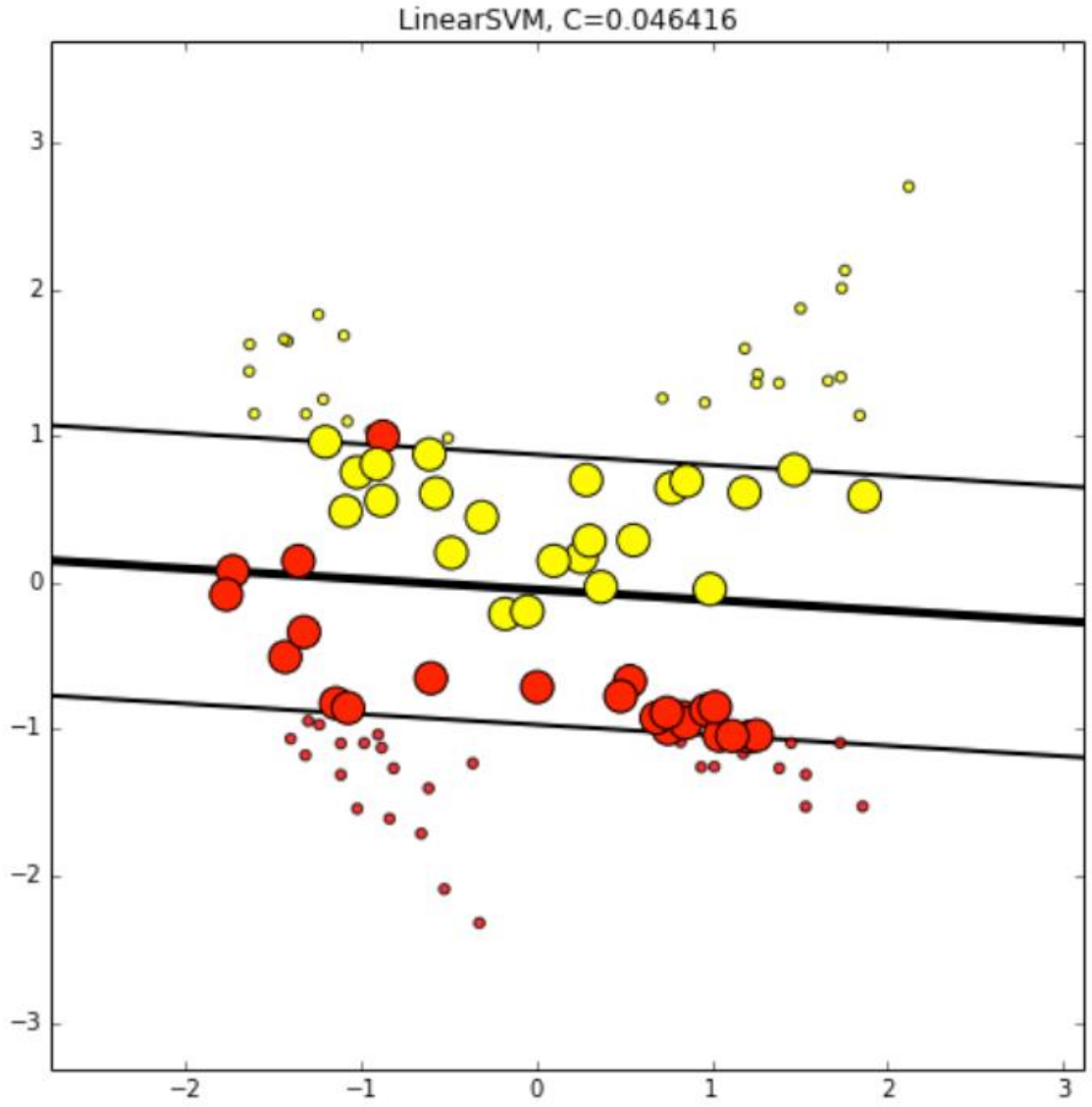
$$\begin{cases} \frac{1}{2}||w||^2 + \boldsymbol{C}\sum_{i=1}^{l}\xi_i \to \min\limits_{w,w_0,\xi_i} (1) \\ y_i\big((w,x_i)+w_0\big) \geq 1-\xi_i, i=1,\dots,l\ (2) \\ \xi_i \geq 0, i=1,\dots,l\ (3) \end{cases}$$

Положительная константа C является управляющим параметром метода и позволяет находить компромисс между максимизацией разделяющей полосы и минимизацией суммарной ошибки.

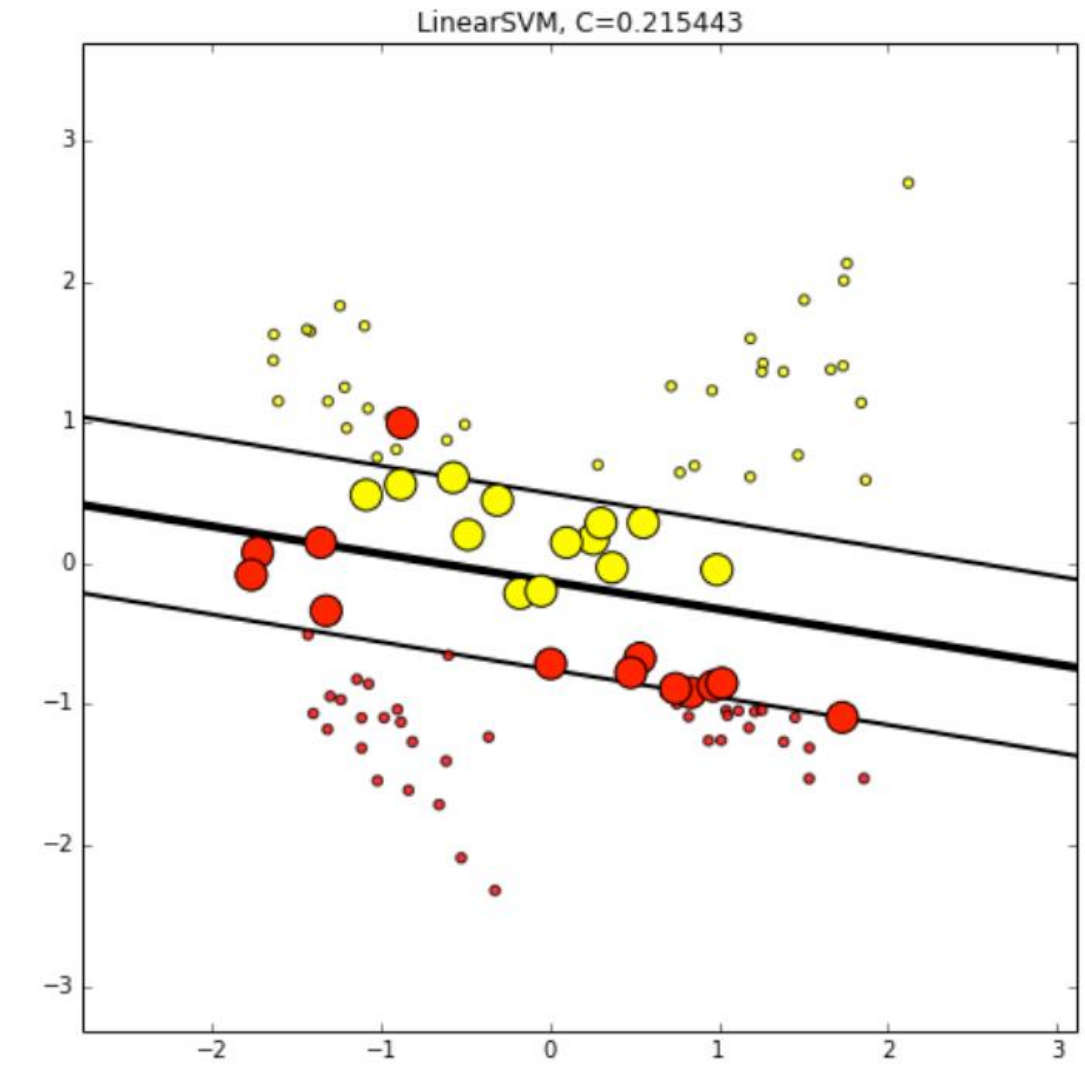
ЗНАЧЕНИЕ КОНСТАНТЫ C



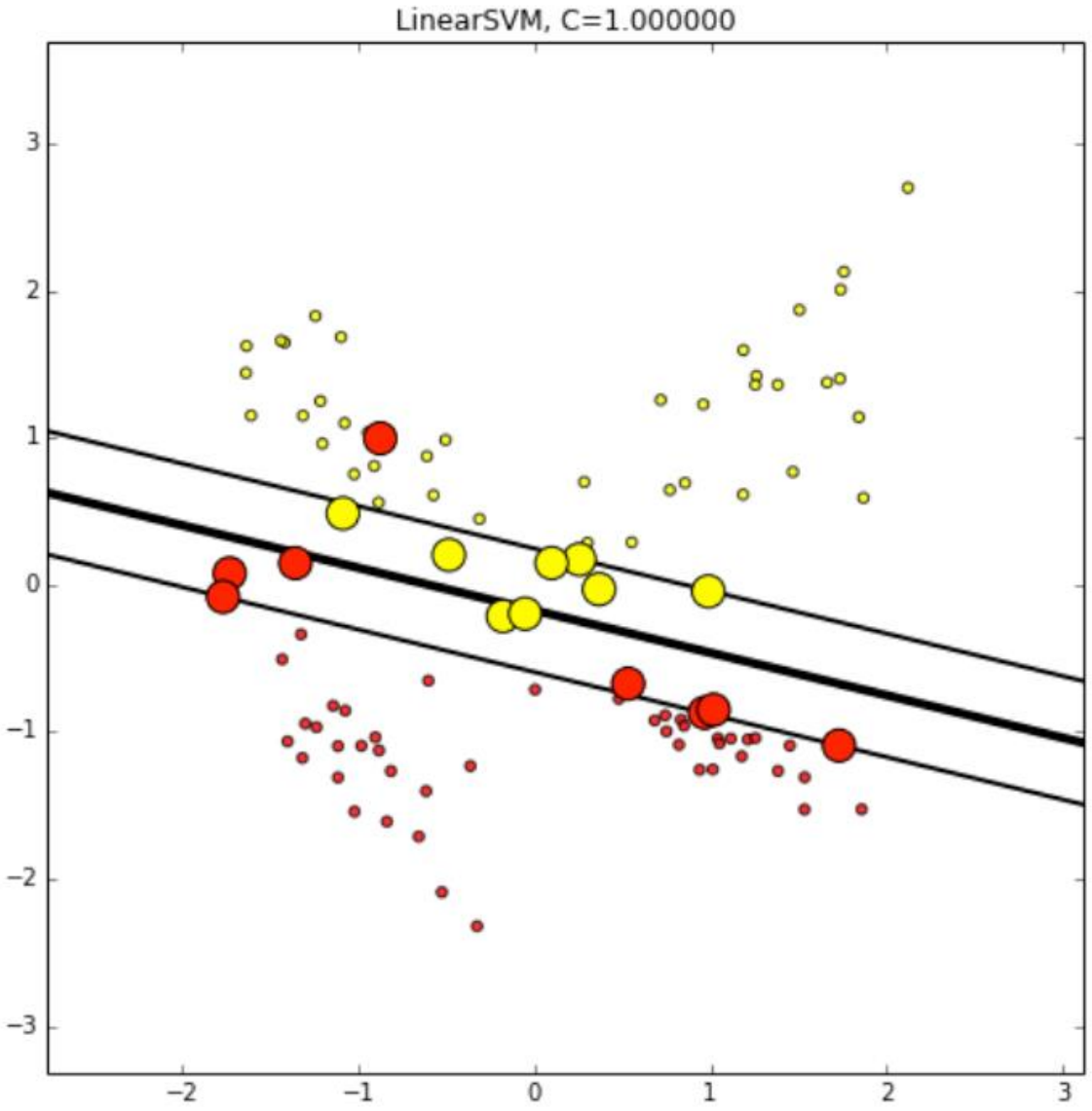
ЗНАЧЕНИЕ КОНСТАНТЫ C



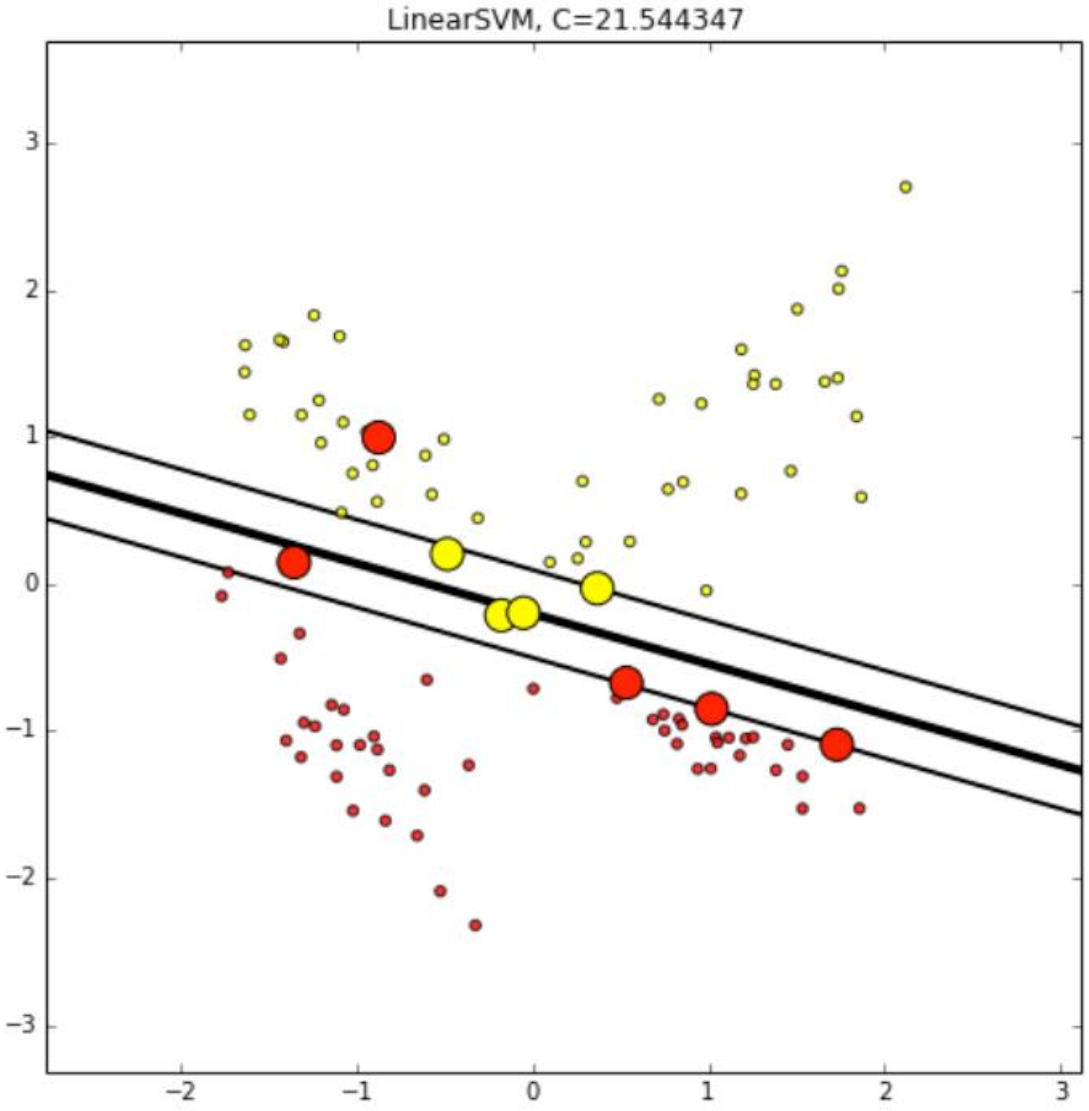
ЗНАЧЕНИЕ КОНСТАНТЫ C



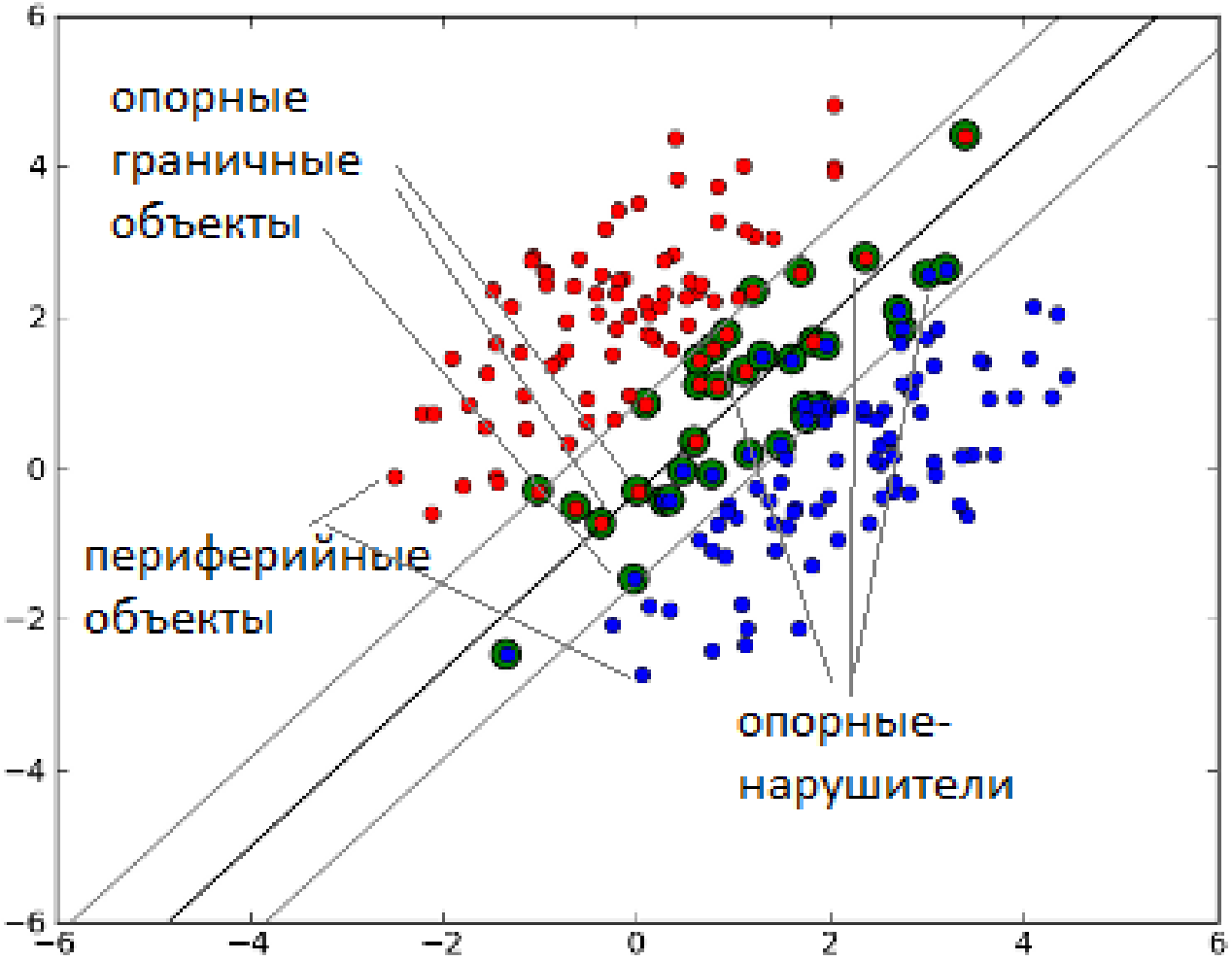
ЗНАЧЕНИЕ КОНСТАНТЫ C



ЗНАЧЕНИЕ КОНСТАНТЫ C



ТИПЫ ОБЪЕКТОВ В SVM

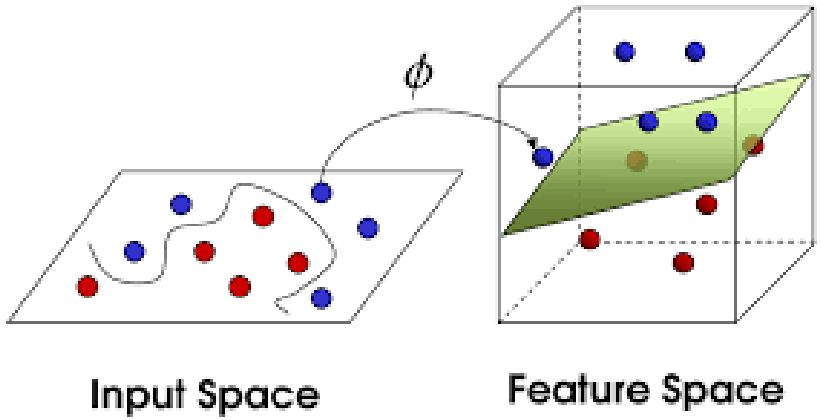


ЯДРОВОЙ МЕТОД ГЛАВНЫХ КОМПОНЕНТ

Пусть исходная выборка (с признаками $x_1, x_2, \dots, x_n$) *линейно не разделима*.

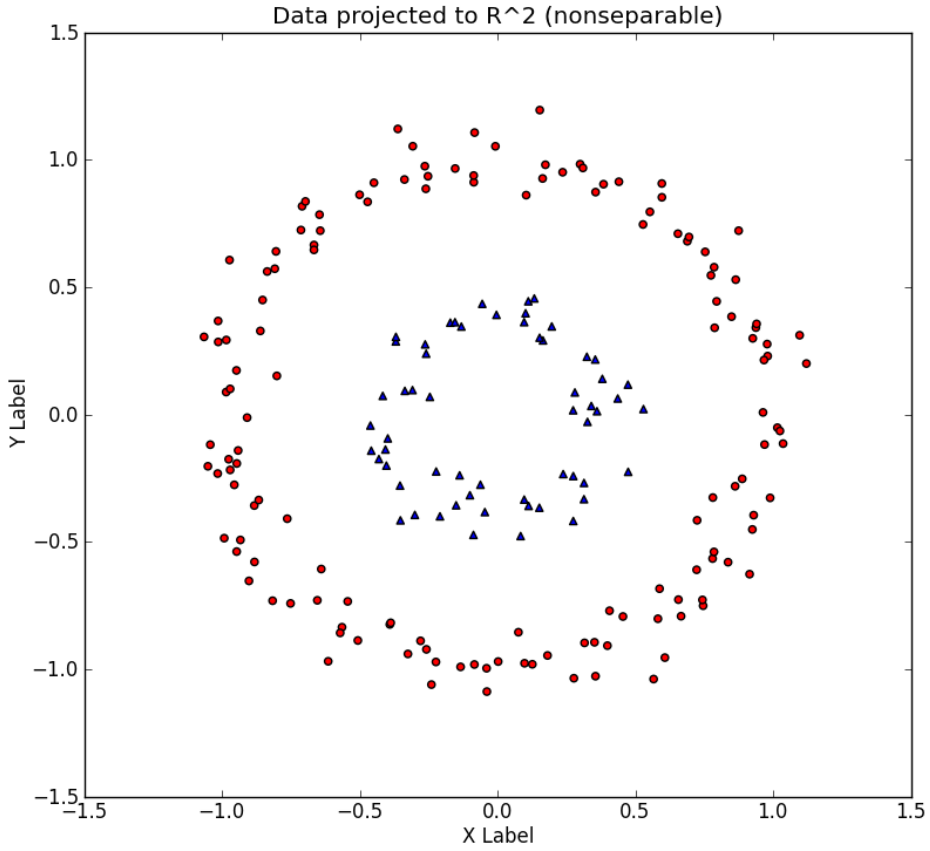
Может существовать такое преобразование координат $(y_1, y_2, \dots, y_N) = f(x_1, x_2, \dots, x_n)$.

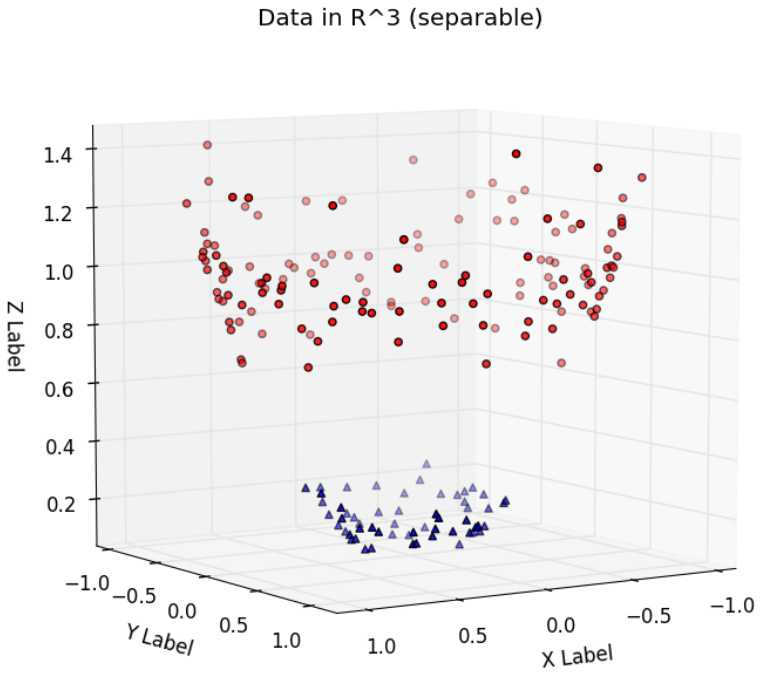
что в пространстве новых координат выборка становится *линейно разделимой*.



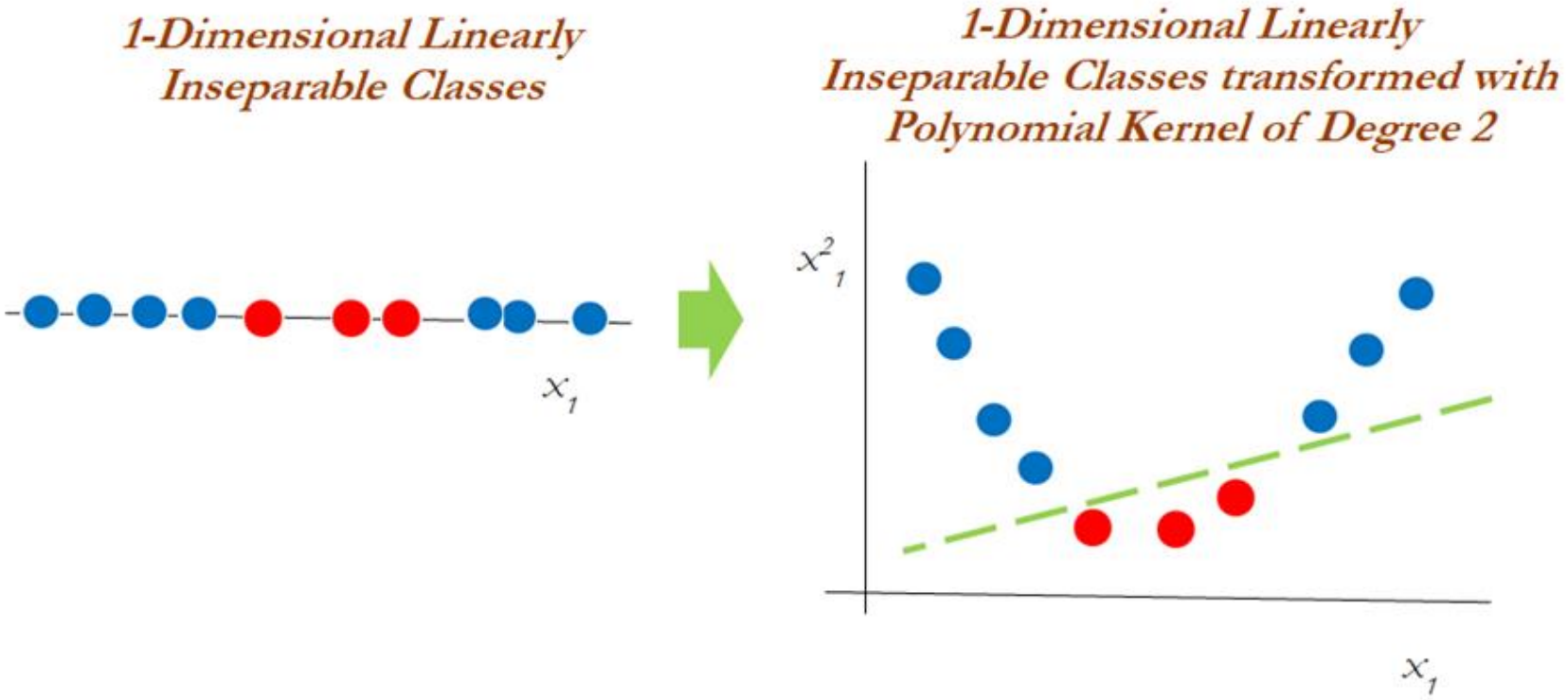
- ***Применение преобразования координат и метода главных компонент называется ядровым методом главных компонент (kernel SVM).***

РАДИАЛЬНОЕ ЯДРО



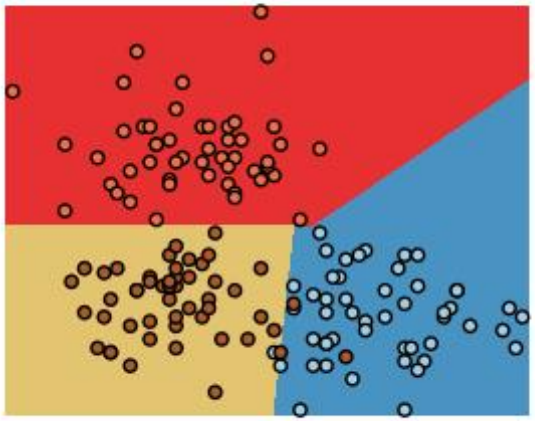


ПОЛИНОМИАЛЬНОЕ ЯДРО

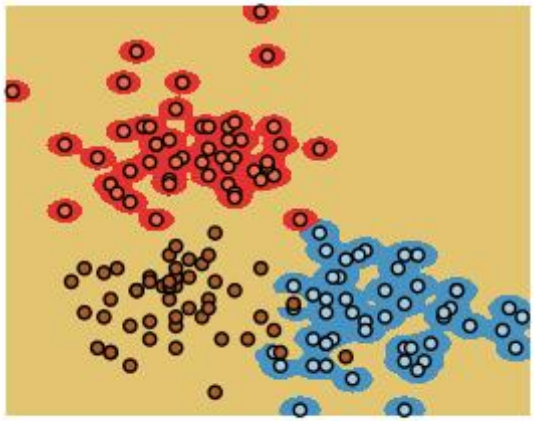


РАЗДЕЛЯЮЩИЕ ПОВЕРХНОСТИ SVM ДЛЯ РАЗЛИЧНЫХ ЯДЕР

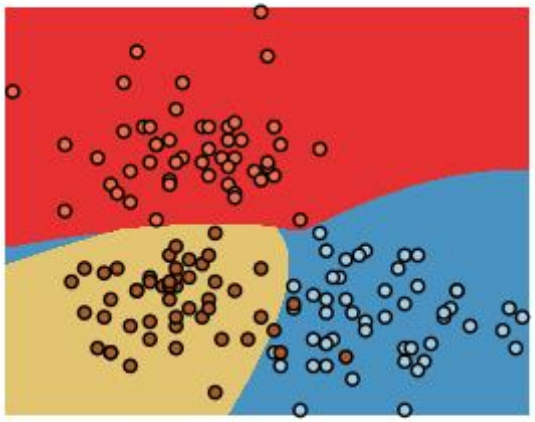
SVC with linear kernel



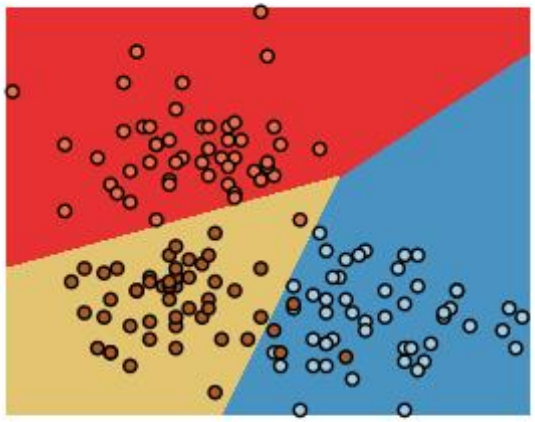
SVC with RBF kernel



SVC with polynomial (degree 3) kernel



LinearSVC (linear kernel)



КАЛИБРОВКА ВЕРОЯТНОСТЕЙ

Калибровка вероятностей - приведение ответов алгоритма к значениям, близким к вероятностям объектов принадлежать конкретному классу.

Зачем это нужно?

- Вероятности гораздо проще интерпретировать
- Вероятности могут дать дополнительную информацию о результатах работы алгоритма

КАЛИБРОВКА ПЛАТТА

- Пусть есть два класса, $Y = \{+1, -1\}$

Задача: для классификатора $a(x)$, предсказывающего значения из отрезка $[0, 1]$, либо предсказывающего класс (+1 или -1), сделать калибровку, чтобы предсказания были вероятностями $p(y = +1|x)$.

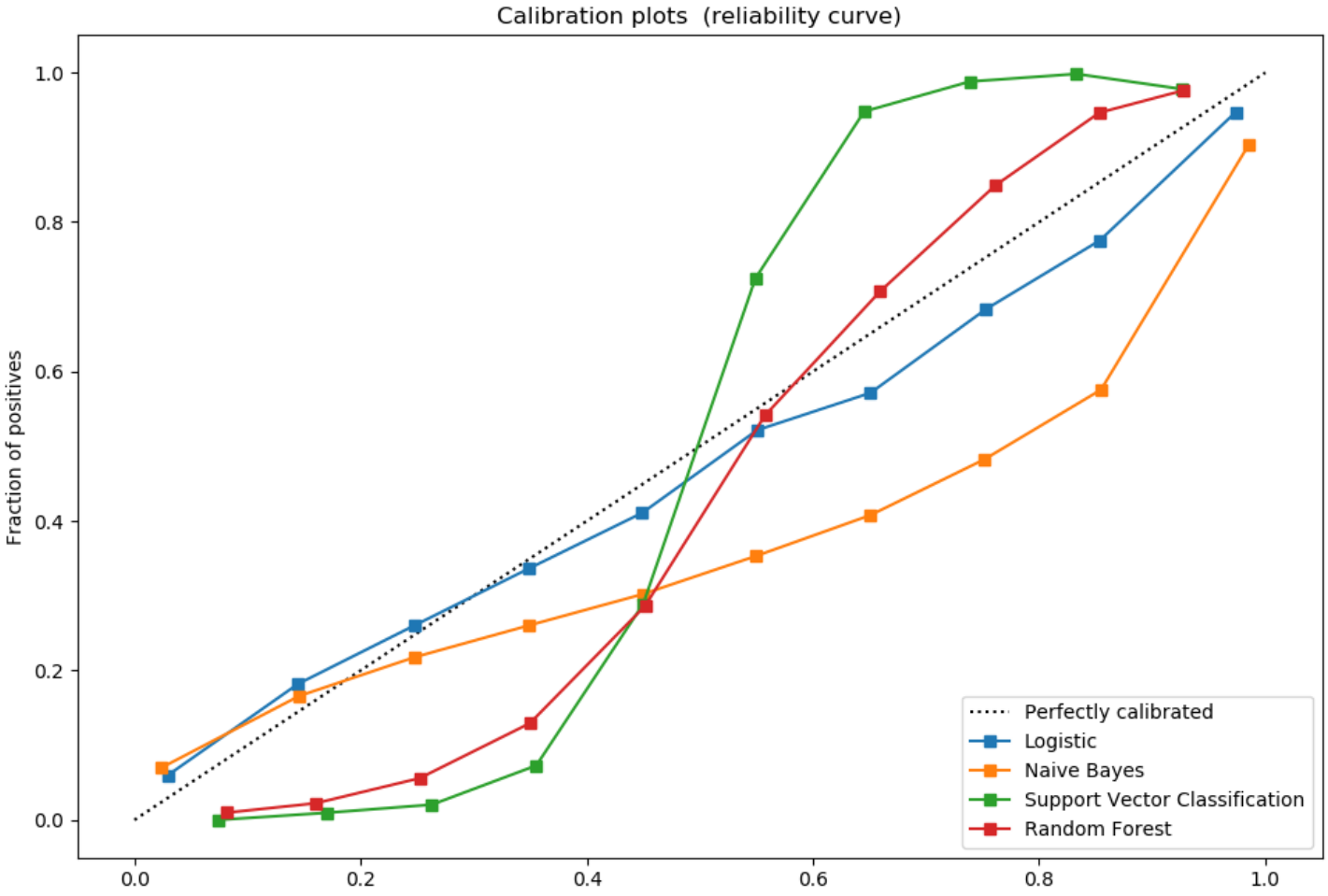
КАЛИБРОВКА ПЛАТТА

- Пусть есть два класса, $Y = \{+1, -1\}$

Задача: для классификатора $a(x)$, предсказывающего значения из отрезка $[0, 1]$, либо предсказывающего класс (+1 или -1), сделать калибровку, чтобы предсказания были вероятностями $p(y = +1|x)$.

Идея: обучаем логистическую регрессию на ответах классификатора $a(x)$.

ПРИМЕР ИЗ SKLEARN



КАЛИБРОВКА ПЛАТТА

- Пусть есть два класса, $Y = \{+1, -1\}$

Задача: для классификатора $a(x)$, предсказывающего значения из отрезка $[0, 1]$, либо предсказывающего класс (+1 или -1), сделать калибровку, чтобы предсказания были вероятностями $p(y = +1|x)$.

Идея: ***обучаем логистическую регрессию на ответах классификатора*** $a(x)$.

- $\pi(x; \alpha; \beta) = \sigma(\alpha \cdot a(x) + \beta) = \frac{1}{1+e^{-(\alpha \cdot a(x)+\beta)}}$
- Находим α и β, минимизируя логистическую функцию потерь:

$$-\sum_{y_i=-1} \log\big(1 - \pi(x; \alpha; \beta)\big) - \sum_{y_i=+1} \log\big(\pi(x; \alpha; \beta)\big) \to \min_{\alpha,\beta}$$