

Лекция 5

Линейные модели классификации. Часть 2.

Кантонистова Е.О.

ВШЭ, 2020

ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

Хотим предсказывать не классы, а вероятности классов.

- Линейная регрессия: $a(x, w) = (x, w) \in \mathbb{R}$
- Логистическая регрессия: $a(x, w) = g((x, w))$,

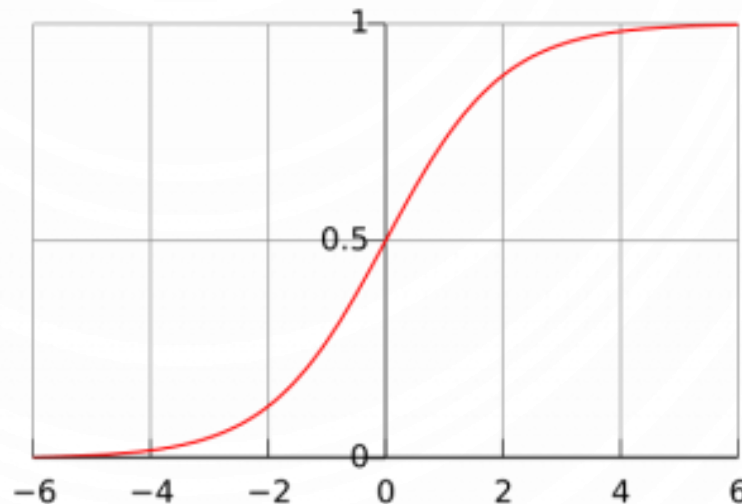
где $g(z) = \frac{1}{1+e^{-z}}$ - сигмоида (логистическая функция)

ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

Хотим предсказывать не классы, а вероятности классов.

- Линейная регрессия: $a(x, w) = (x, w) \in \mathbb{R}$
- Логистическая регрессия: $a(x, w) = g((x, w))$,

где $g(z) = \frac{1}{1+e^{-z}}$ - сигмоида (логистическая функция),
 $g(z) \in (0; 1)$.



Логистическая регрессия: $a(x, w) = \frac{1}{1+e^{-(x,w)}}$

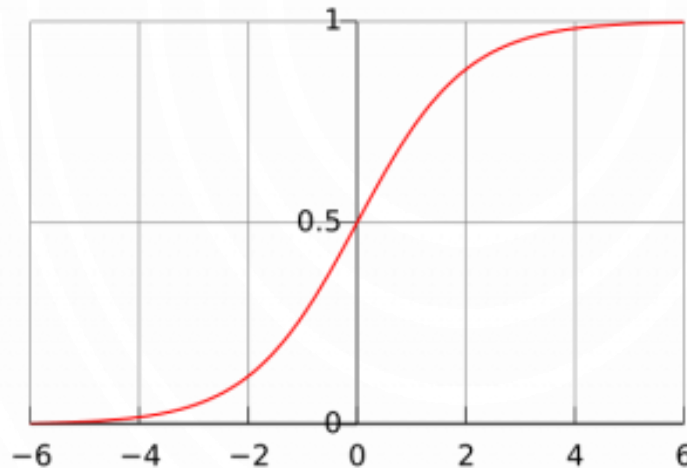
ВЕРОЯТНОСТНЫЙ СМЫСЛ

- $a(x, w)$ – вероятность того, что $y = +1$ на объекте x , то есть

$$a(x, w) = P(y = +1|x; w)$$

РАЗДЕЛЯЮЩАЯ ГРАНИЦА

Предсказываем $y = +1$, если $a(x, w) \geq 0.5$.



$a(x, w) = g((x, w)) \geq 0.5$, если $(x, w) \geq 0$.

Получаем, что

- $y = +1$ при $(x, w) \geq 0$
- $y = -1$ при $(x, w) < 0$,

т.е. $(x, w) = 0$ – разделяющая гиперплоскость.

ЛОГИСТИЧЕСКАЯ РЕГРЕССИЯ

Логистическая регрессия - это линейный классификатор!

ФУНКЦИЯ ПОТЕРЬ ЛОГИСТИЧЕСКОЙ РЕГРЕССИИ

Если взять квадратичную функцию потерь

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

то возникнут проблемы:

- $Q(a, X) = \frac{1}{l} \sum_{i=1}^l \left(\frac{1}{1+e^{-(x,w)}} - y \right)^2$ - не выпуклая функция
(*можем не попасть в глобальный минимум при оптимизации*)

ФУНКЦИЯ ПОТЕРЬ ЛОГИСТИЧЕСКОЙ РЕГРЕССИИ

Если взять квадратичную функцию потерь

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

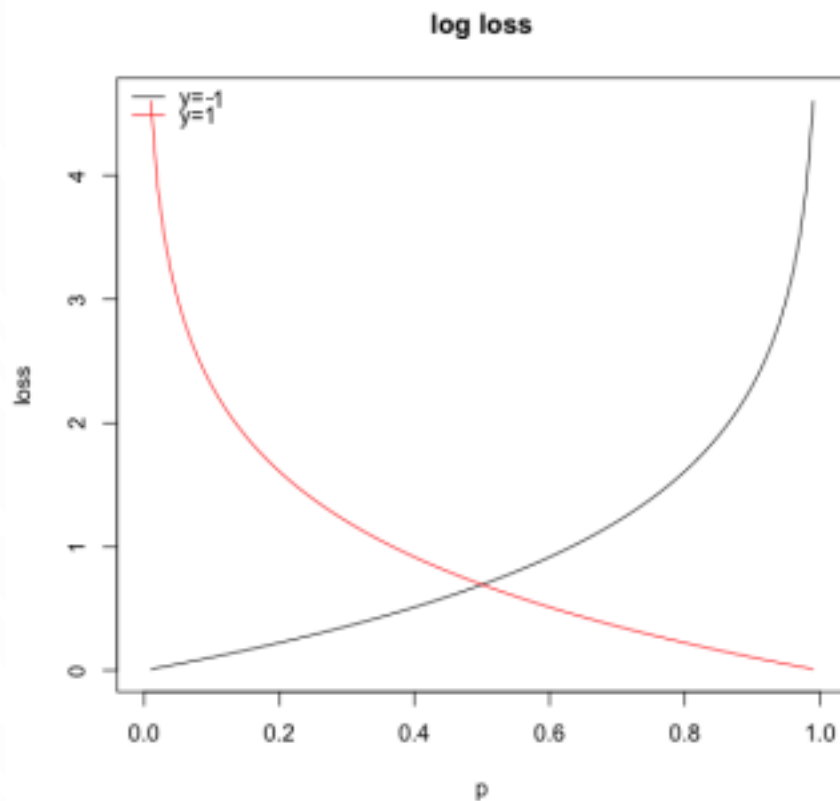
то возникнут проблемы:

- $Q(a, X) = \frac{1}{l} \sum_{i=1}^l \left(\frac{1}{1+e^{-(x,w)}} - y \right)^2$ - не выпуклая функция
(*можем не попасть в глобальный минимум при оптимизации*)
- *На совсем неправильном предсказании маленький штраф* (пусть предсказали вероятность 0% на объекте класса $y = +1$, тогда штраф всего $(1 - 0)^2 = 1$)

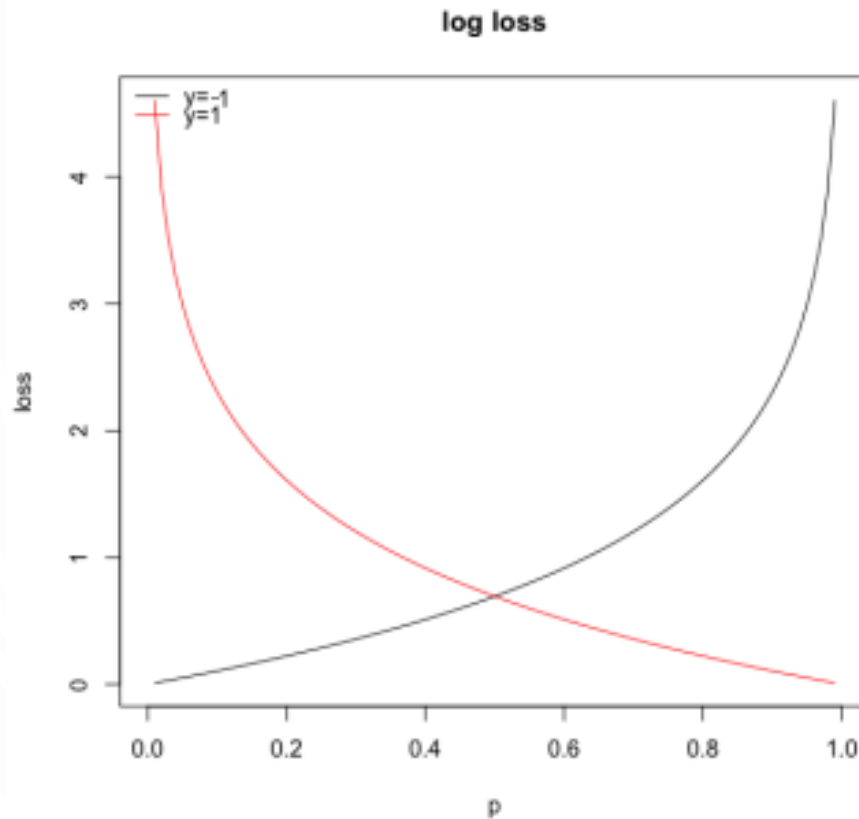
ФУНКЦИЯ ПОТЕРЬ ЛОГИСТИЧЕСКОЙ РЕГРЕССИИ

Возьмем логистическую функцию потерь (log-loss):

$$Q(w) = - \sum_{i=1}^l ([y_i = +1] \cdot \log(a(x_i, w)) + [y_i = -1] \cdot \log(1 - a(x_i, w)))$$



ЛОГИСТИЧЕСКАЯ ФУНКЦИЯ ПОТЕРЬ



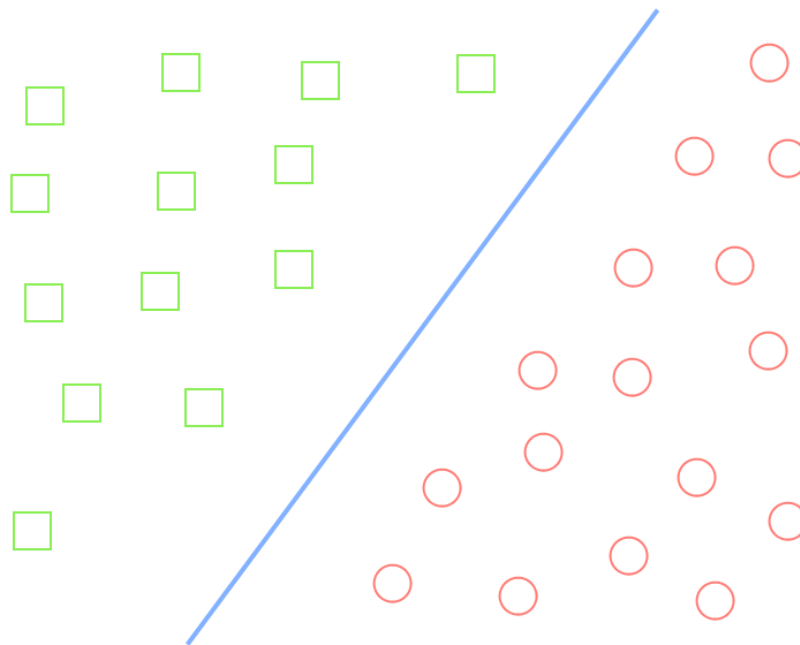
- если $a(x, w) = 1$ и $y = +1$, то штраф $L(a, y) = 0$
- если $a(x, w) \rightarrow 0$, а $y = +1$, то штраф $L(a, y) \rightarrow +\infty$

The image features a light gray background with a subtle pattern of concentric circles. In the four corners, there are decorative circuit-like lines in a light blue color. These lines consist of straight segments and small circles, resembling a stylized electronic circuit or a network diagram. The lines are more prominent in the top-left and bottom-left corners, and more sparse in the top-right and bottom-right corners.

МЕТОД ОПОРНЫХ ВЕКТОРОВ

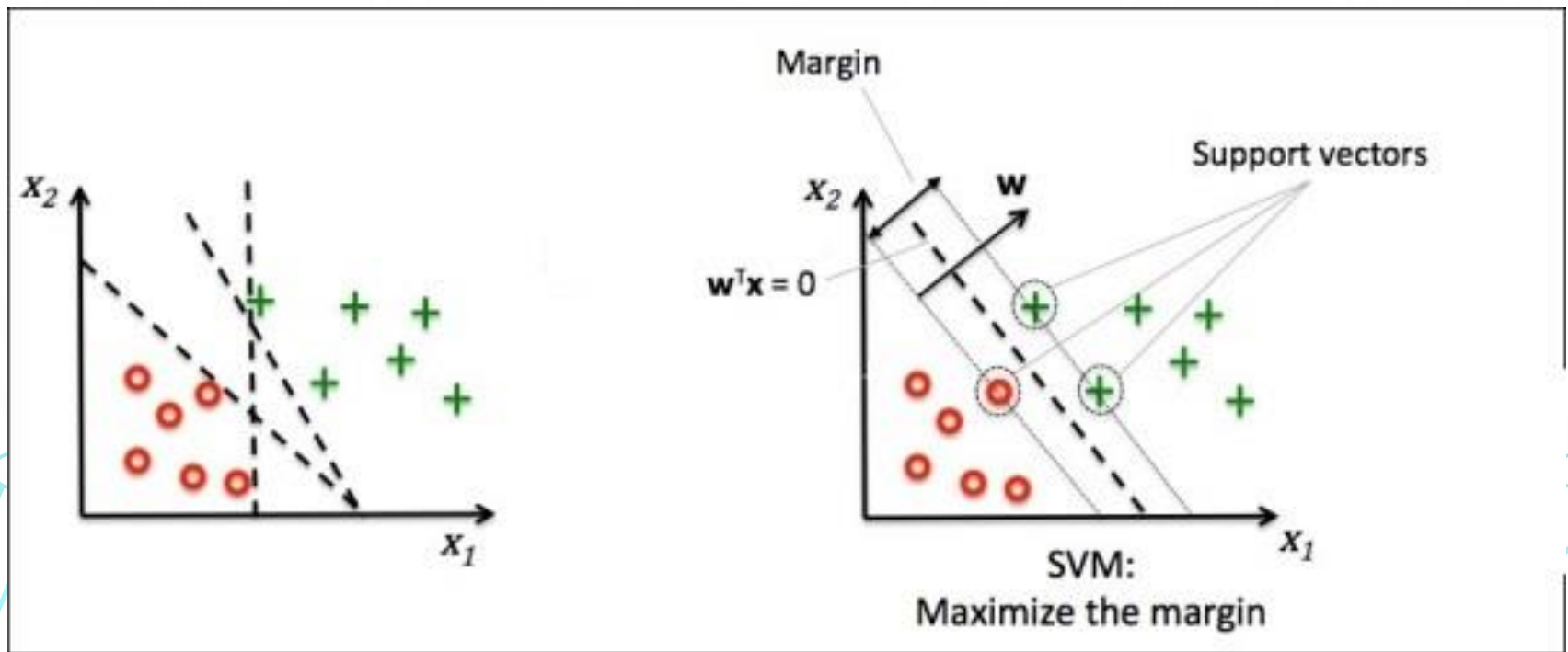
ЛИНЕЙНО РАЗДЕЛИМАЯ ВЫБОРКА

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

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

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

МЕТОД ОПОРНЫХ ВЕКТОРОВ: РАЗДЕЛИМЫЙ СЛУЧАЙ

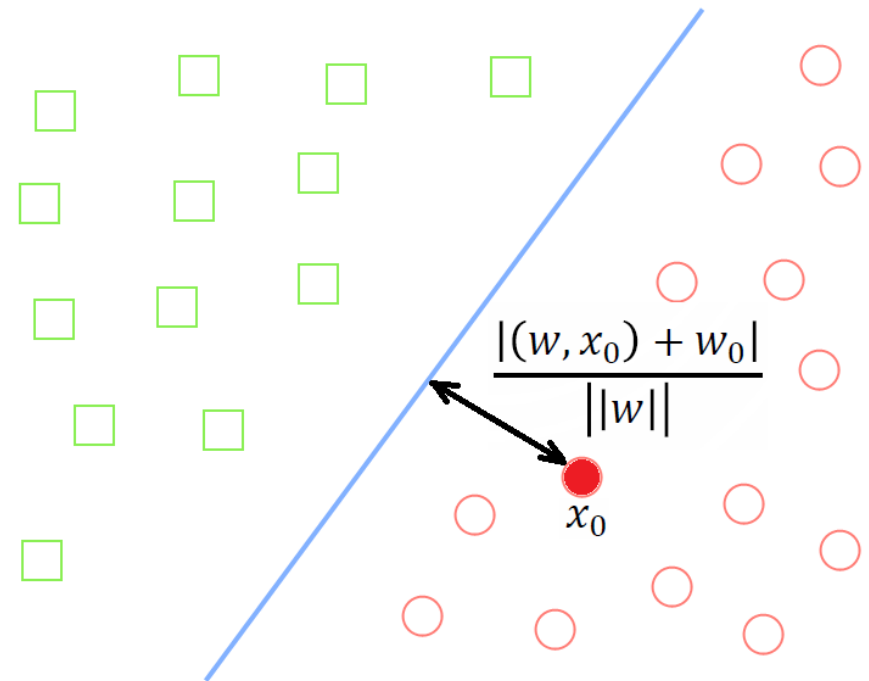
- $a(x) = \text{sign}((w, x) + w_0)$
- Нормируем параметры w и w_0 так, что

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

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

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

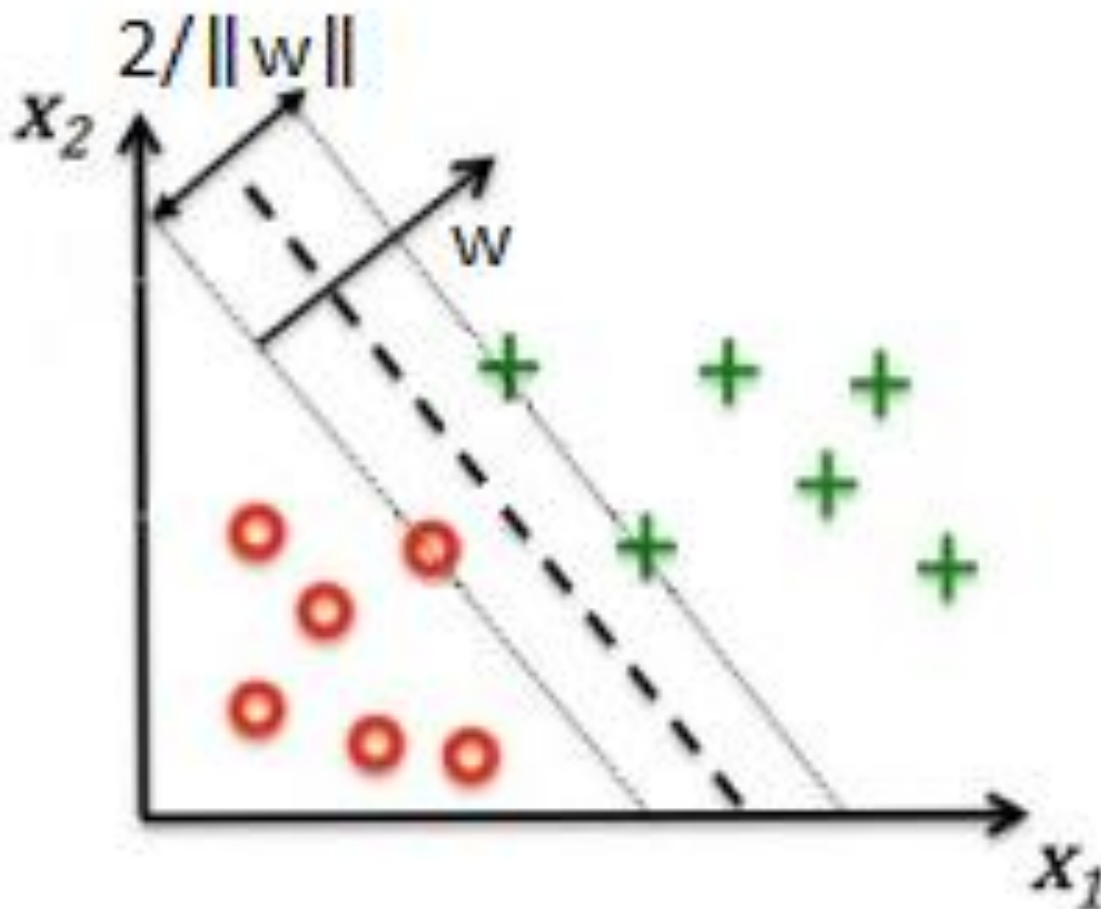
Тогда расстояние от точки x_0 до разделяющей гиперплоскости, задаваемой классификатором:

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

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

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

РАЗДЕЛЯЮЩАЯ ПОЛОСА



ОПТИМИЗАЦИОННАЯ ЗАДАЧА SVM ДЛЯ РАЗДЕЛИМОЙ ВЫБОРКИ

$$\begin{cases} \frac{1}{2} \|w\|^2 \rightarrow \min_w \\ y_i((w, x_i) + w_0) \geq 1, i = 1, \dots, l \end{cases}$$

Утверждение. Данная оптимизационная задача имеет единственное решение.

ЛИНЕЙНО НЕРАЗДЕЛИМАЯ ВЫБОРКА

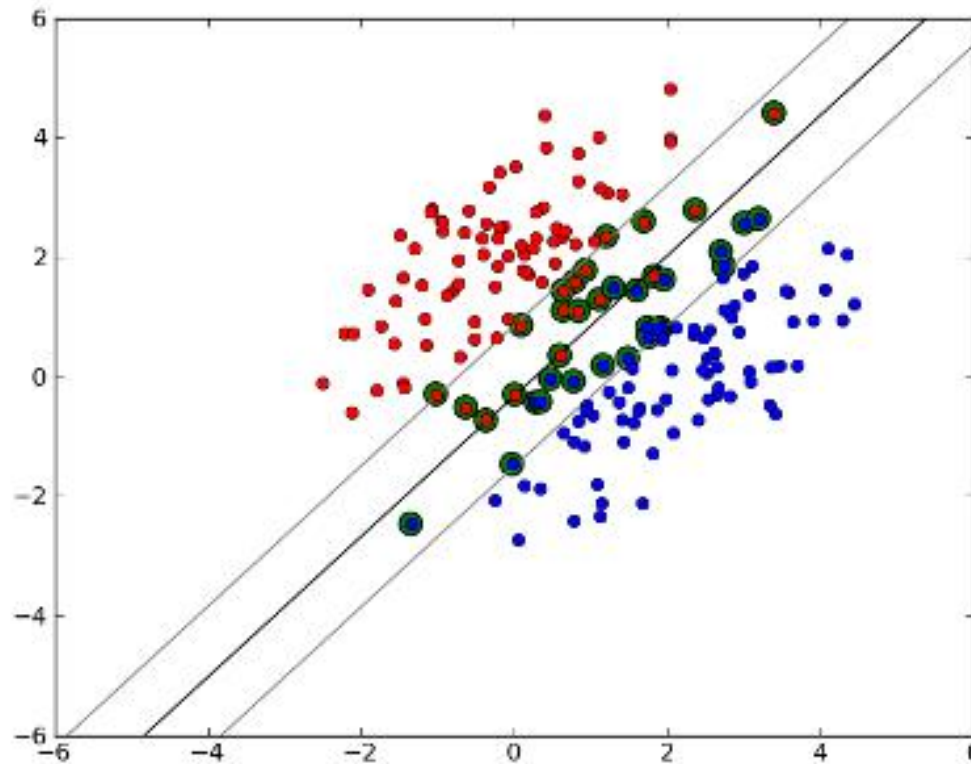
- Существует хотя бы один объект $x \in X$, что

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

ЛИНЕЙНО НЕРАЗДЕЛИМАЯ ВЫБОРКА

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

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



ЛИНЕЙНО НЕРАЗДЕЛИМАЯ ВЫБОРКА

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

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

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

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

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:

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

МЕТОД ОПОРНЫХ ВЕКТОРОВ: НЕРАЗДЕЛИМЫЙ СЛУЧАЙ

Утверждение. Задача

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

Является выпуклой и имеет единственное решение.

СВЕДЕНИЕ К БЕЗУСЛОВНОЙ ЗАДАЧЕ

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

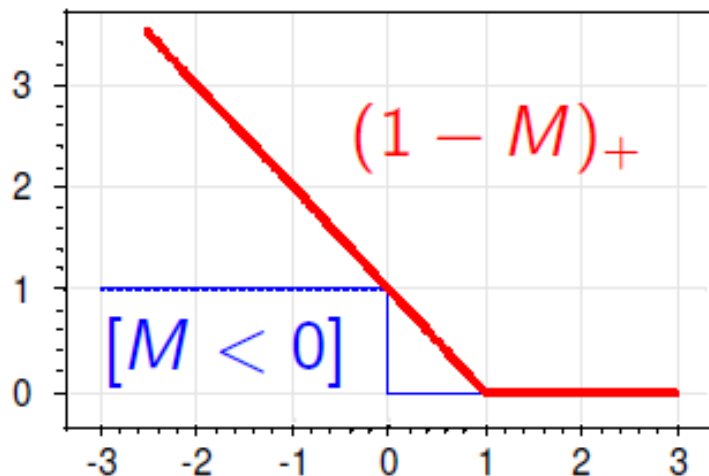
Эту задачу можно переписать в другом, более простом виде:

$$\frac{1}{2} \|w\|^2 + C \sum_{i=1}^l \max(0, 1 - y_i((w, x_i) + w_0)) \rightarrow \min_{w, w_0}$$

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:

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

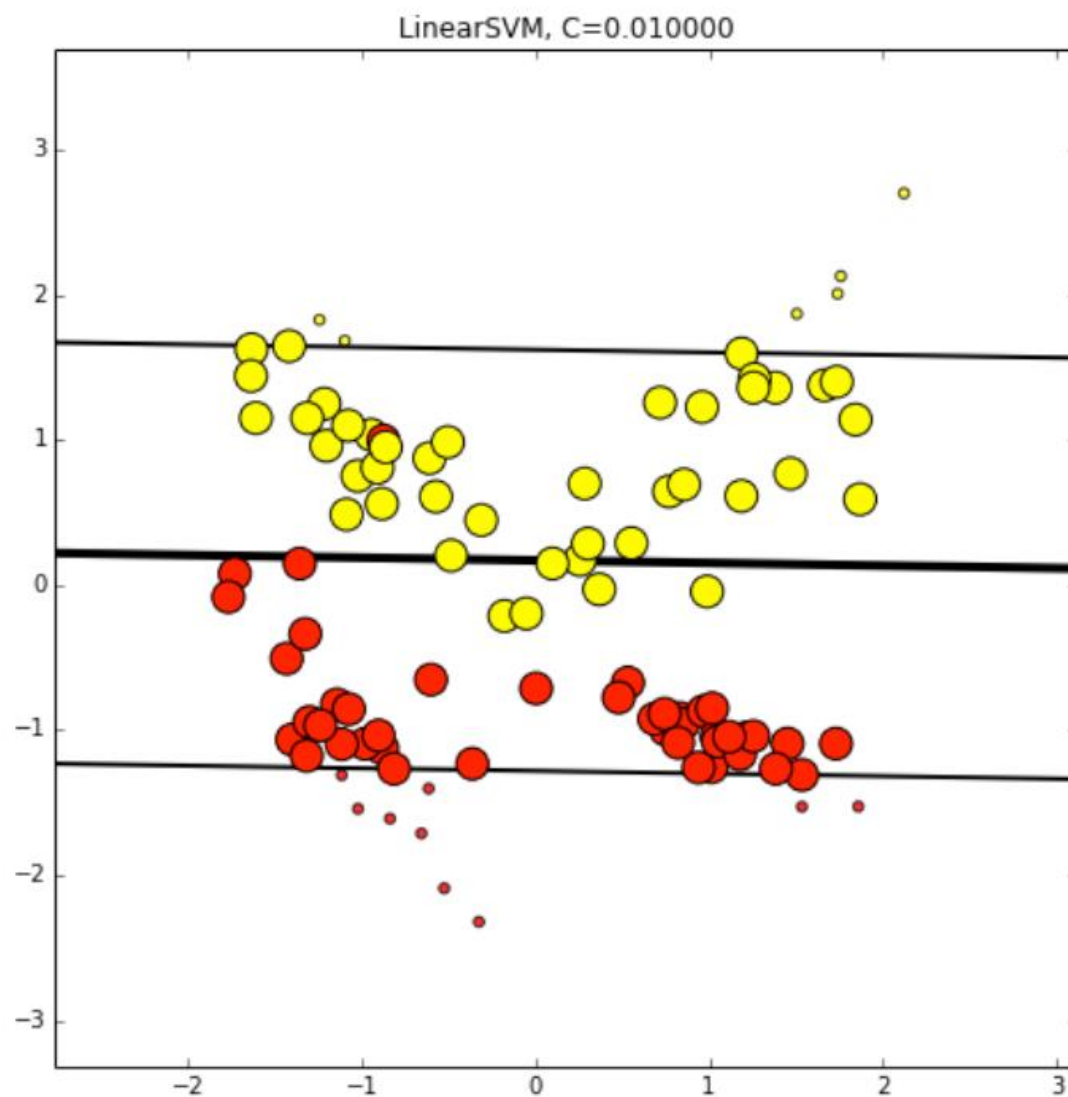


ЗНАЧЕНИЕ КОНСТАНТЫ C

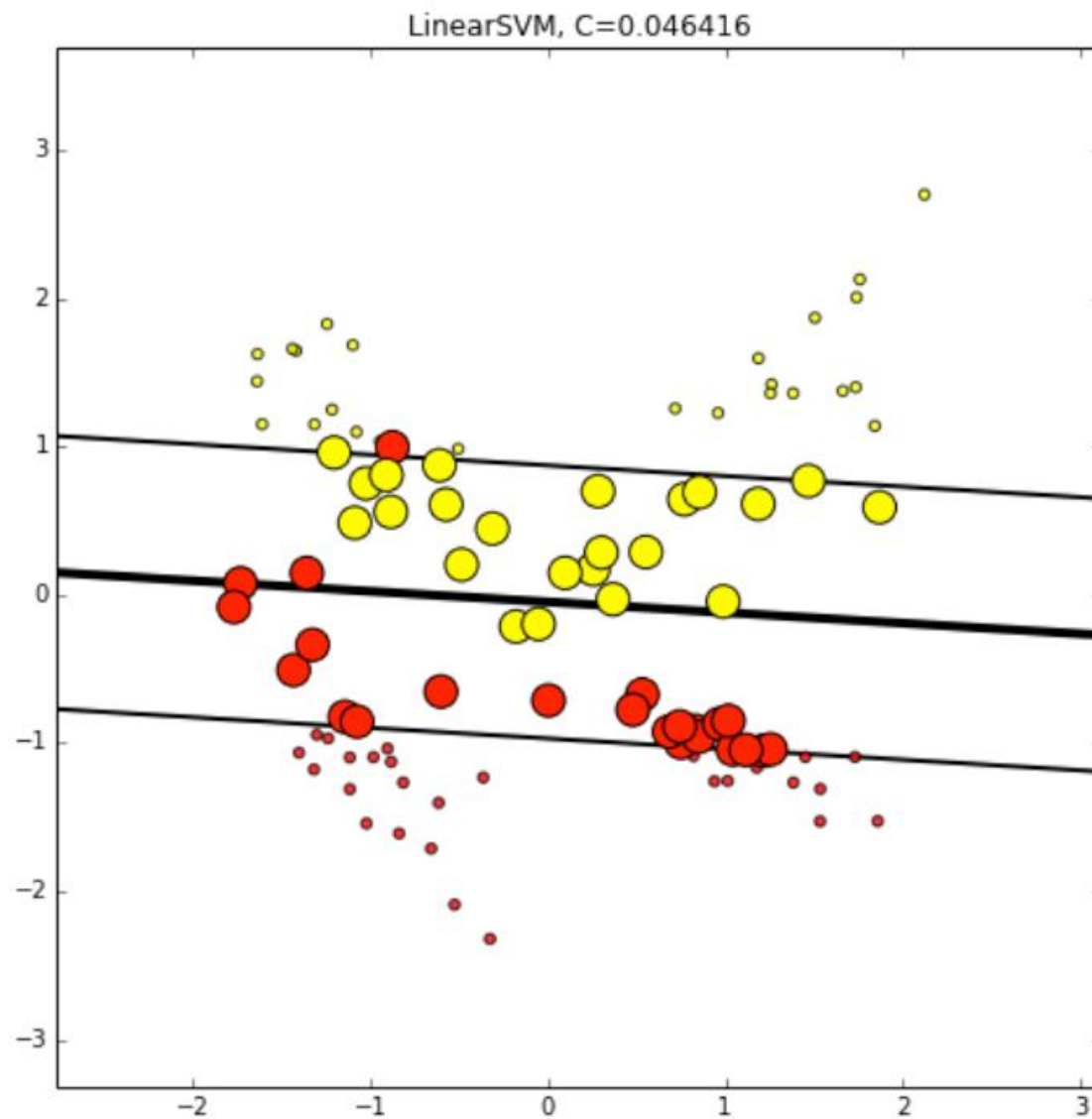
$$\begin{cases} \frac{1}{2} \|w\|^2 + \textcolor{red}{C} \sum_{i=1}^l \xi_i \rightarrow \min_{w, w_0, \xi_i} (1) \\ y_i((w, x_i) + w_0) \geq 1 - \xi_i, i = 1, \dots, l (2) \\ \xi_i \geq 0, i = 1, \dots, l (3) \end{cases}$$

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

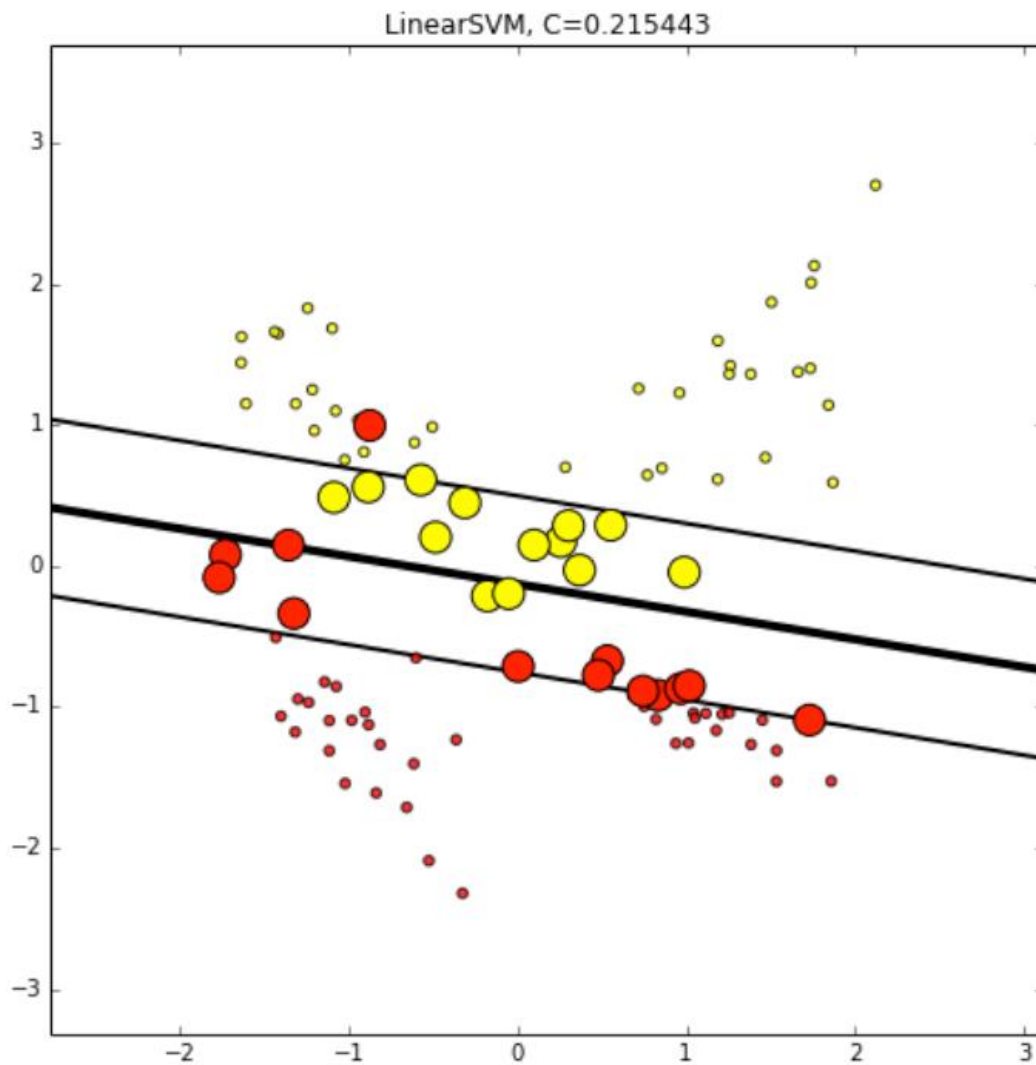
ЗНАЧЕНИЕ КОНСТАНТЫ C



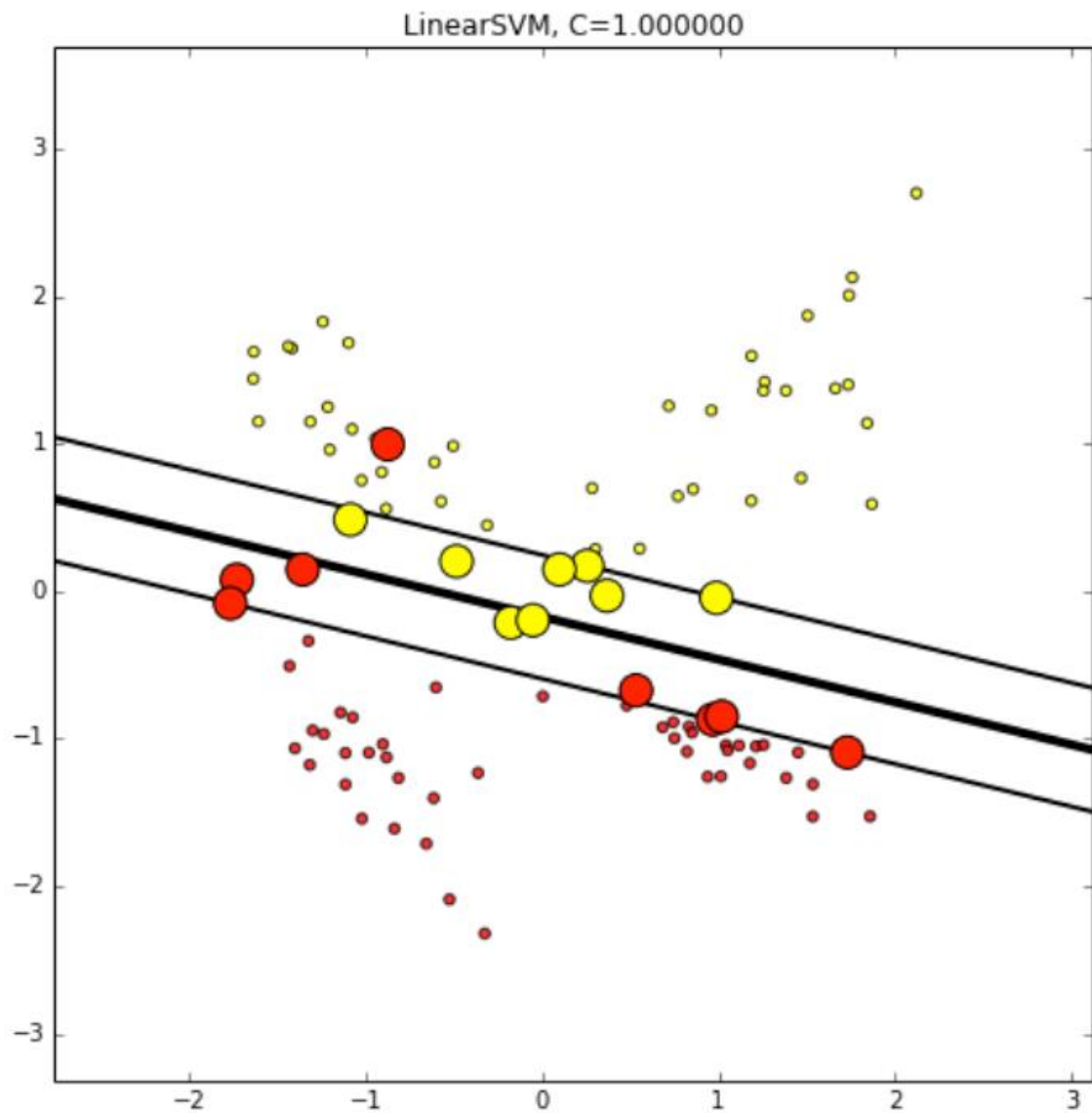
ЗНАЧЕНИЕ КОНСТАНТЫ C



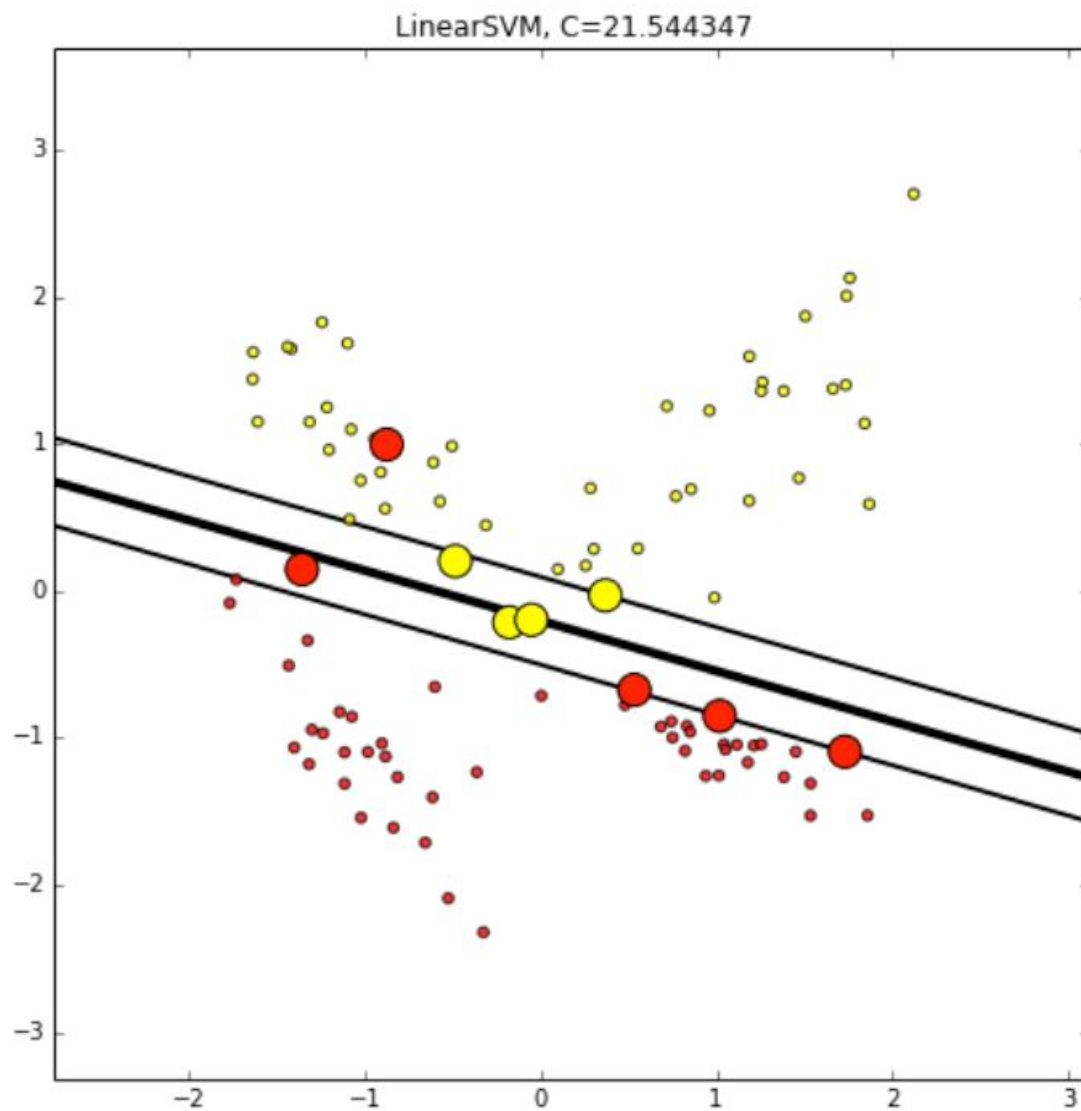
ЗНАЧЕНИЕ КОНСТАНТЫ C



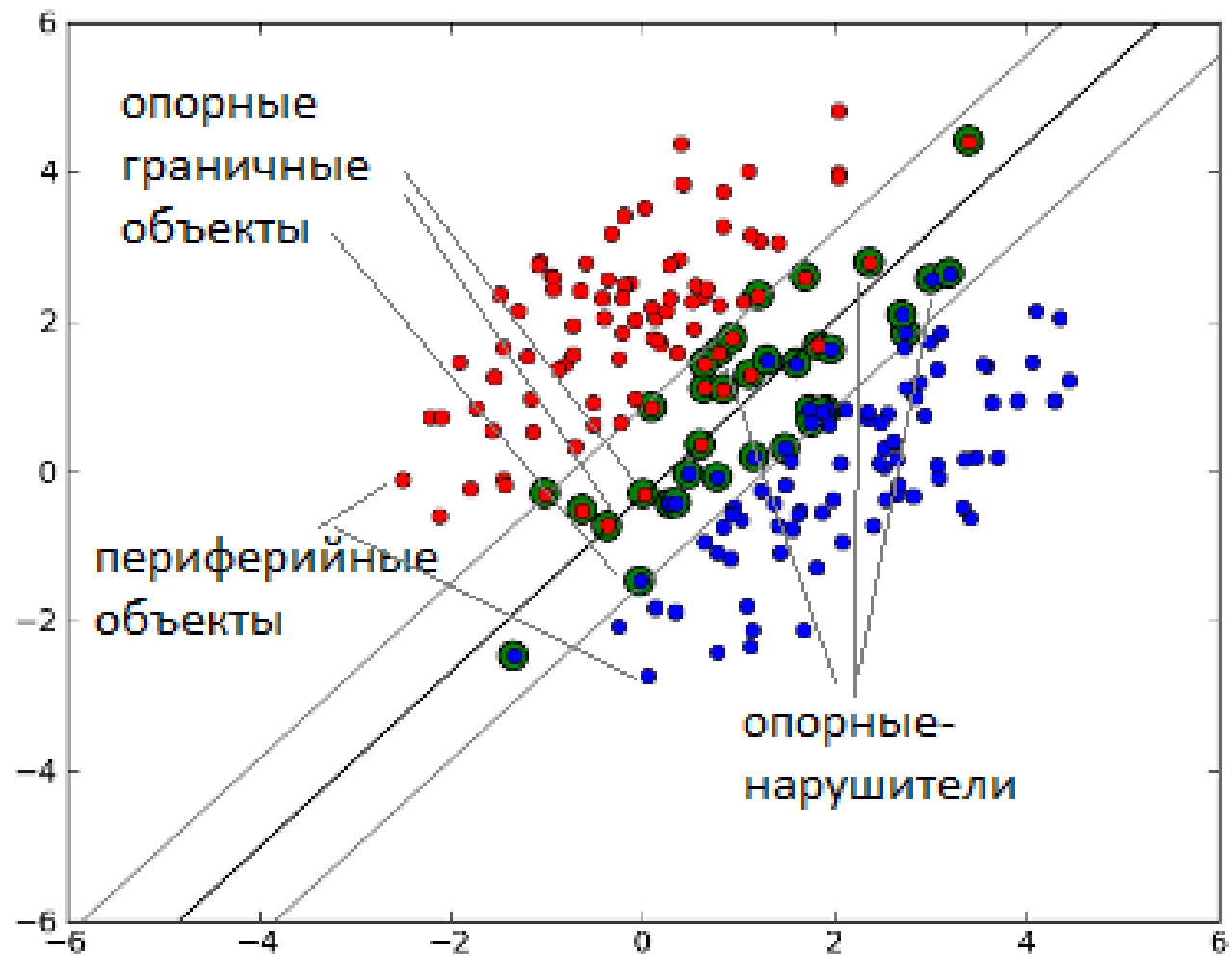
ЗНАЧЕНИЕ КОНСТАНТЫ C



ЗНАЧЕНИЕ КОНСТАНТЫ C



ТИПЫ ОБЪЕКТОВ В SVM

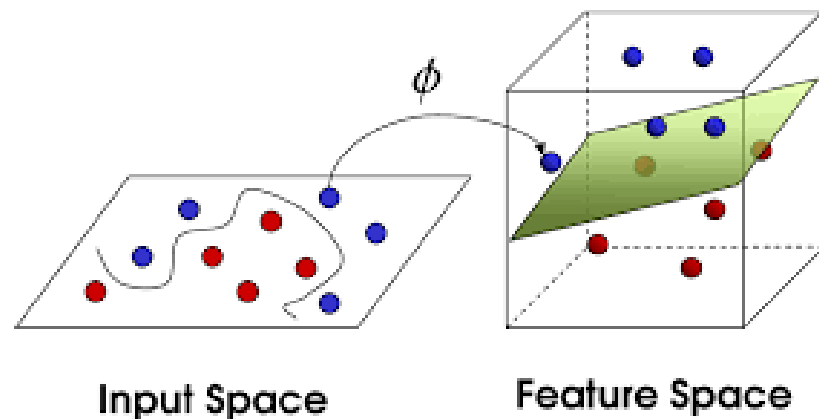


ЯДРОВОЙ МЕТОД ОПОРНЫХ ВЕКТОРОВ

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

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

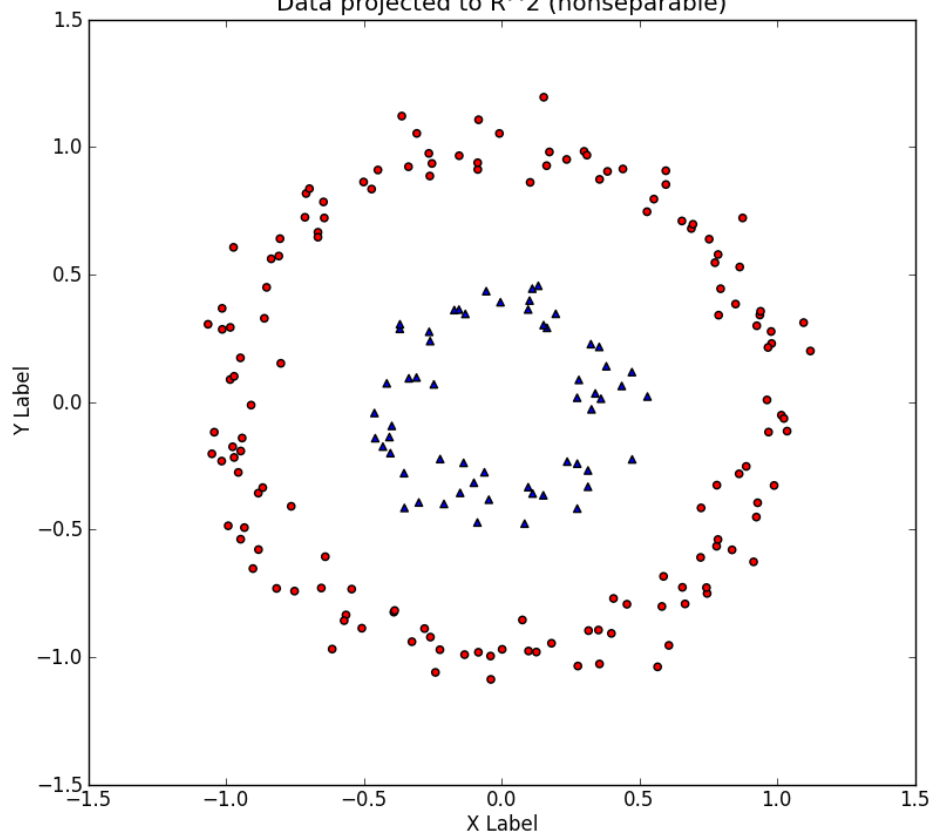
что в пространстве новых координат выборка становится *линейно разделимой*.



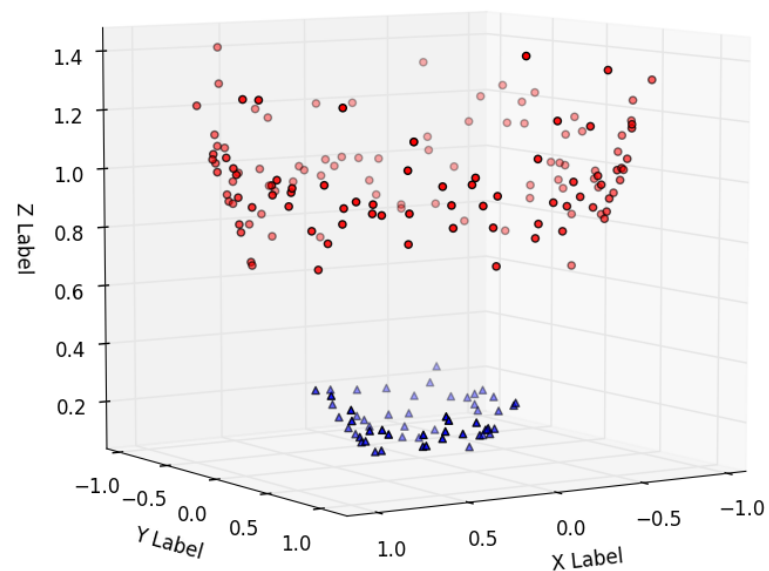
- **Применение преобразования координат и метода главных компонент называется ядровым методом опорных векторов (kernel SVM).**

РАДИАЛЬНОЕ ЯДРО

Data projected to R^2 (nonseparable)

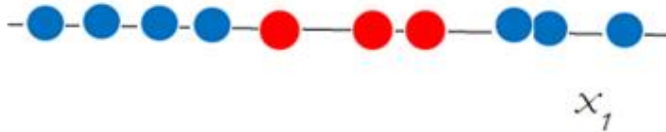


Data in R^3 (separable)

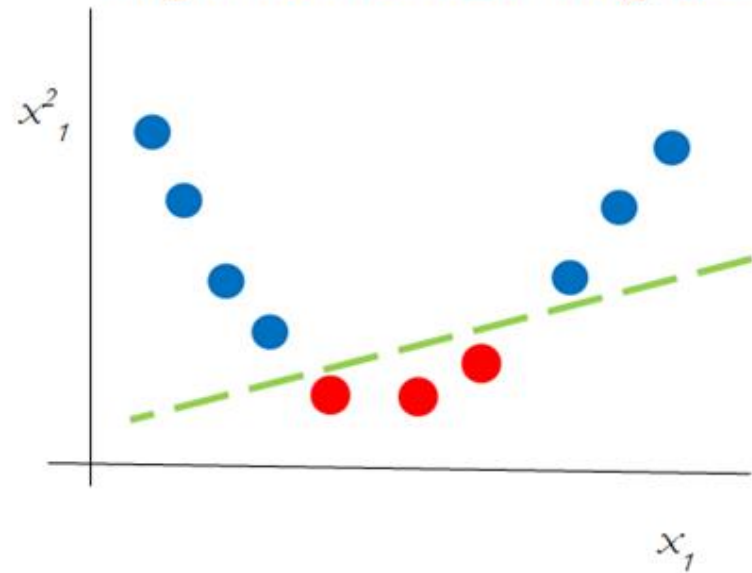


ПОЛИНОМИАЛЬНОЕ ЯДРО

*1-Dimensional Linearly
Inseparable Classes*

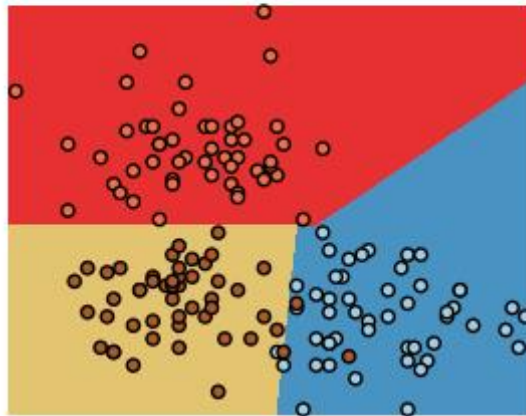


*1-Dimensional Linearly
Inseparable Classes transformed with
Polynomial Kernel of Degree 2*

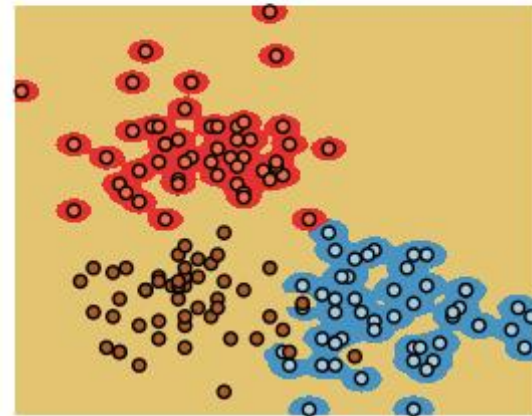


РАЗДЕЛЯЮЩИЕ ПОВЕРХНОСТИ SVM ДЛЯ РАЗЛИЧНЫХ ЯДЕР

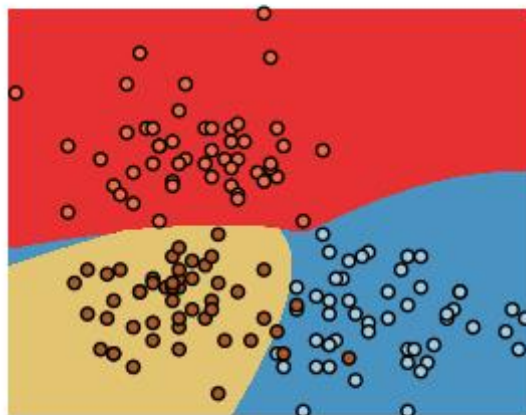
SVC with linear kernel



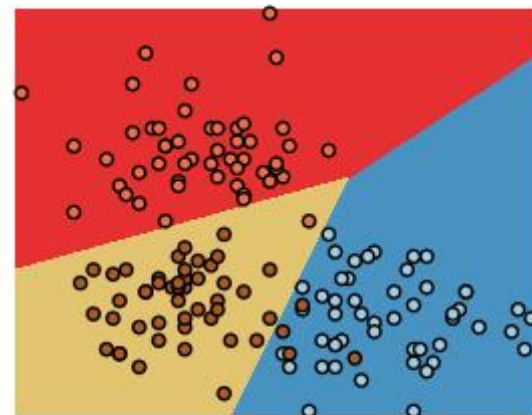
SVC with RBF kernel



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 $p(y = +1|x)$.

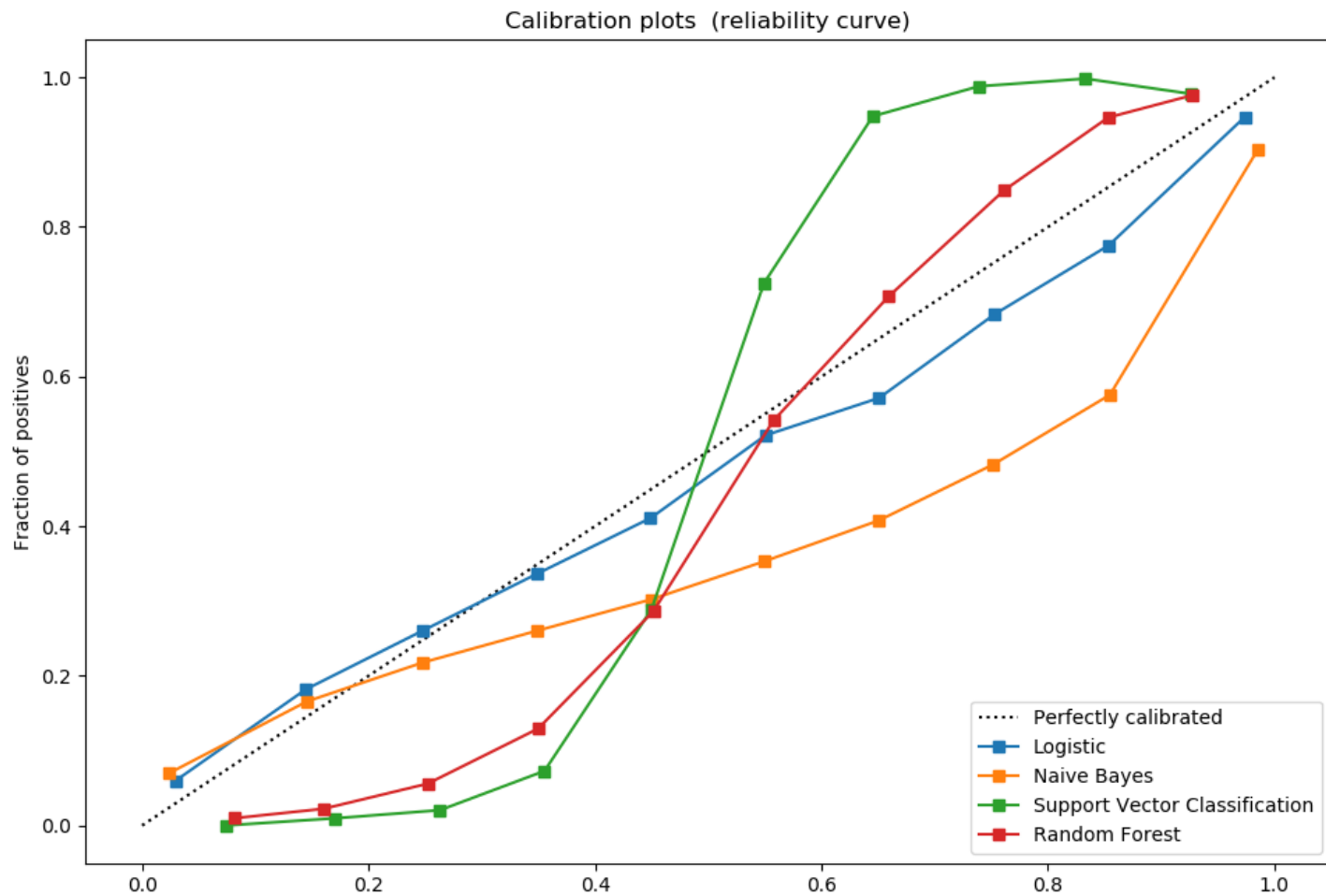
КАЛИБРОВКА ПЛАТТА

- 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 $a(x)$.*

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 $a(x)$.*

- $$\pi(x; \alpha; \beta) = \sigma(\alpha \cdot a(x) + \beta) = \frac{1}{1 + e^{-(\alpha \cdot a(x) + \beta)}}$$

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, *то есть обучаем логистическую регрессию.*

*Обучаем логистическую регрессию мы не сами, а пользуясь моделью CalibratedClassifier в sklearn.