Линейная классификация

Лекция 3

План лекции

- Линейные модели классификации
- Метрики качества классификации
- Логистическая регрессия
- Обучение логистической регрессии
- Метод опорных векторов
- Многоклассовая классификация

Линейные модели классификации

- Обозначим:
  - $\mathbb{X} = \mathbb{R}^d$ – пространство объектов
  - $Y = \{-1, +1\}$ – множество допустимых ответов
  - $X = \{(\vec{x}_i, y_i)\}_{i=1}^{l}$ – обучающая выборка
- Линейная модель классификации:

$$a(\vec{x}) = sign(\langle \vec{w}, \vec{x} \rangle + w_0) = sign\left(\sum_{j=1}^{d} w_j x_j + w_0\right),$$

где $\vec{w} \in \mathbb{R}^d$ – вектор весов, $w_0 \in \mathbb{R}$ – сдвиг, $sign$ – функция знака:

$$sign\, u = \begin{cases} +1, u > 0 \\ 0, u = 0 \\ -1, u < 0 \end{cases}$$

- Если $x_0 = 1$, тогда $a(\vec{x}) = sign\langle \vec{w}, \vec{x} \rangle$

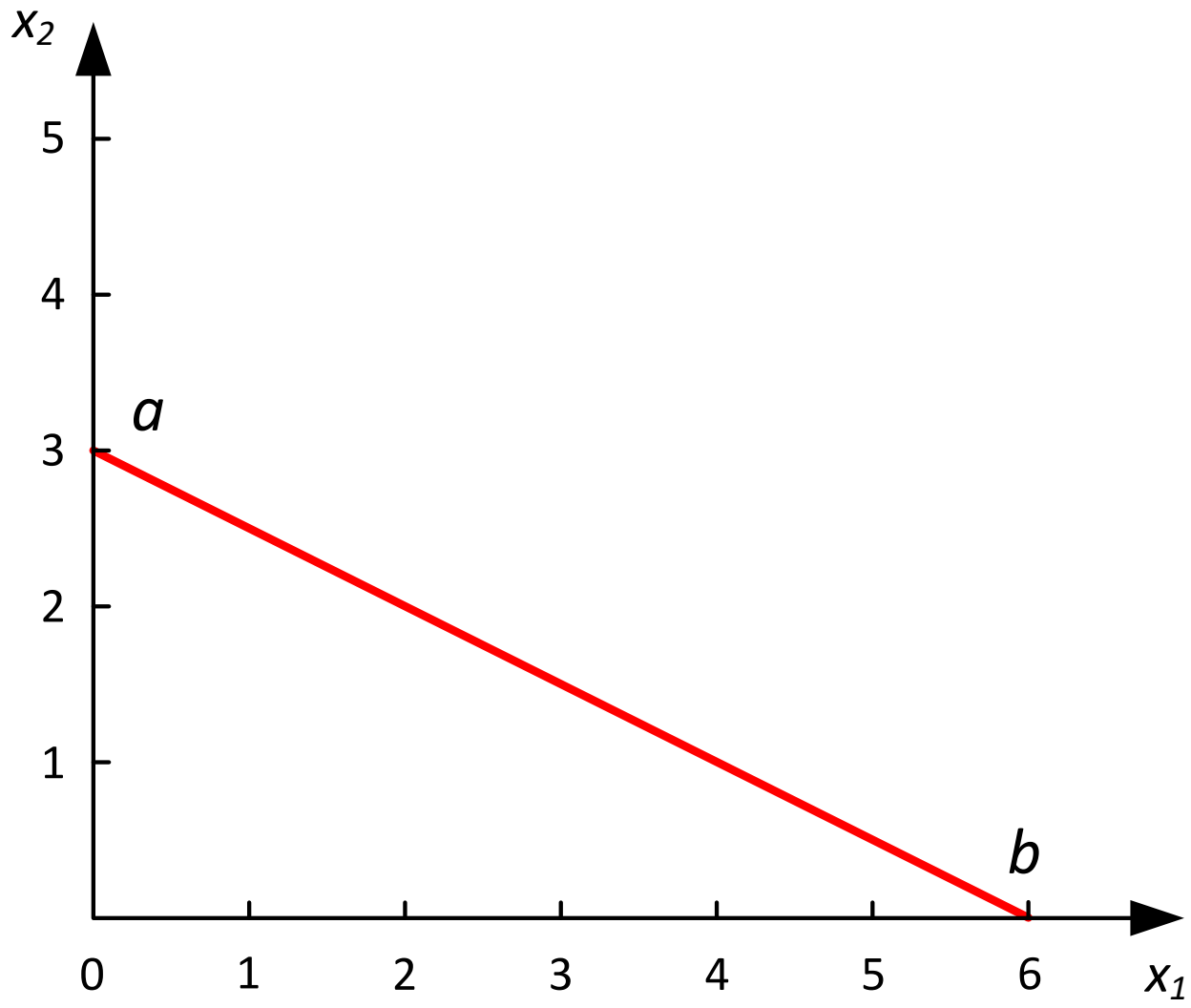
Геометрическая интерпретация

- Линейный классификатор соответствует гиперплоскости с вектором нормали $\vec{w}$
- Величина скалярного произведения $\langle \vec{w}, \vec{x} \rangle$ пропорциональна расстоянию от гиперплоскости до точки $\vec{x}$, а его знак показывает, с какой стороны от гиперплоскости находится данная точка
- Расстояние от точки до гиперплоскости:

$$\frac{|\langle \vec{w}, \vec{x} \rangle|}{\|\vec{w}\|}$$

- Линейный классификатор разделяет пространство на две части с помощью гиперплоскости, и при этом одно полупространство относится к положительному классу, а другое – к отрицательному

Геометрическая интерпретация



Геометрическая интерпретация

- Уравнение прямой по двум точкам:

$$(x_{1a}, x_{2a}) = (0, 3), \qquad (x_{1b}, x_{2b}) = (6, 0)$$

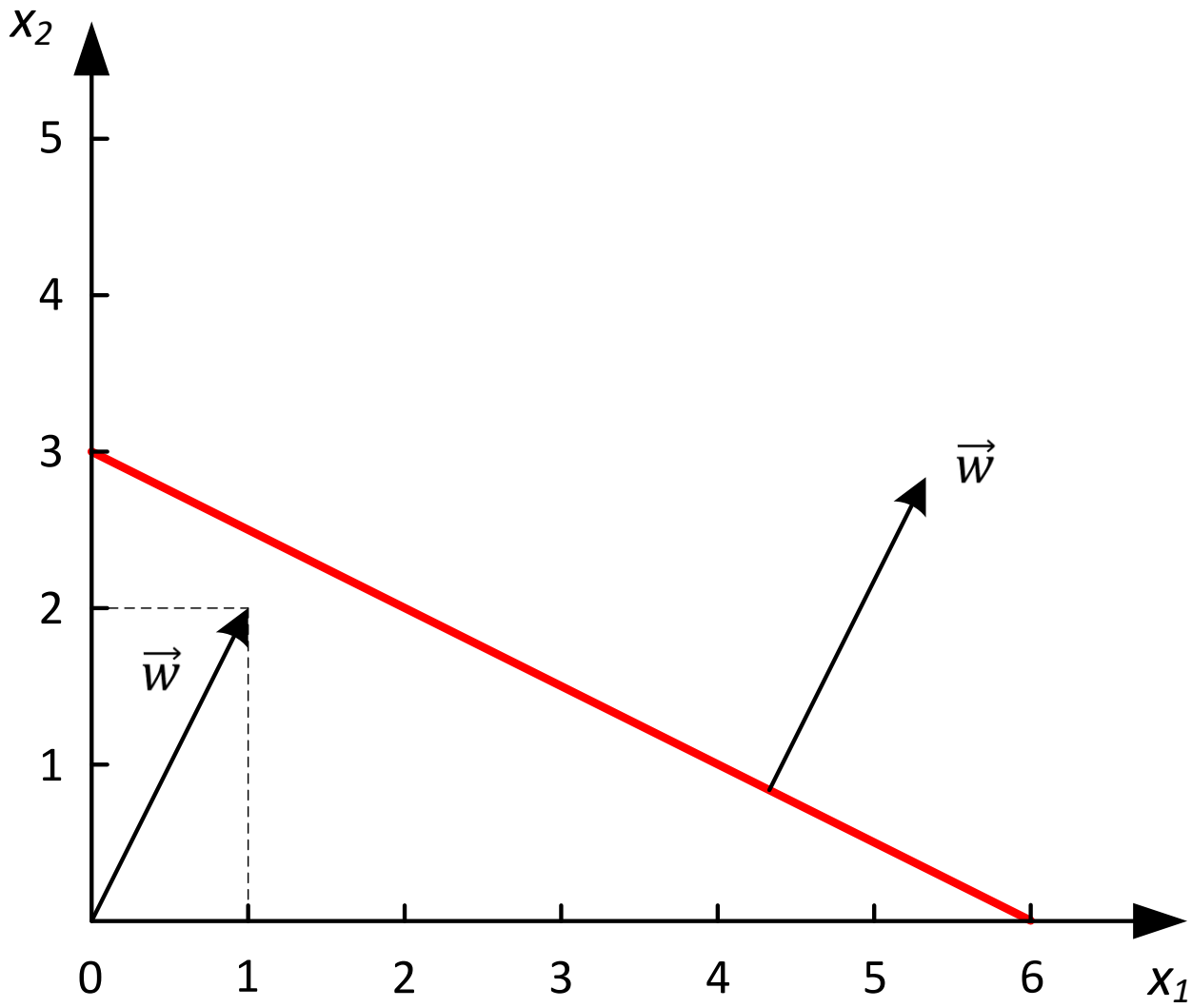
$$(x_{2a} - x_{2b})x_1 + (x_{1b} - x_{1a})x_2 + (x_{1a}x_{2b} - x_{1b}x_{2a}) = 0$$

$$3x_1 + 6x_2 - 18 = 0$$

$$x_1 + 2x_2 - 6 = 0$$

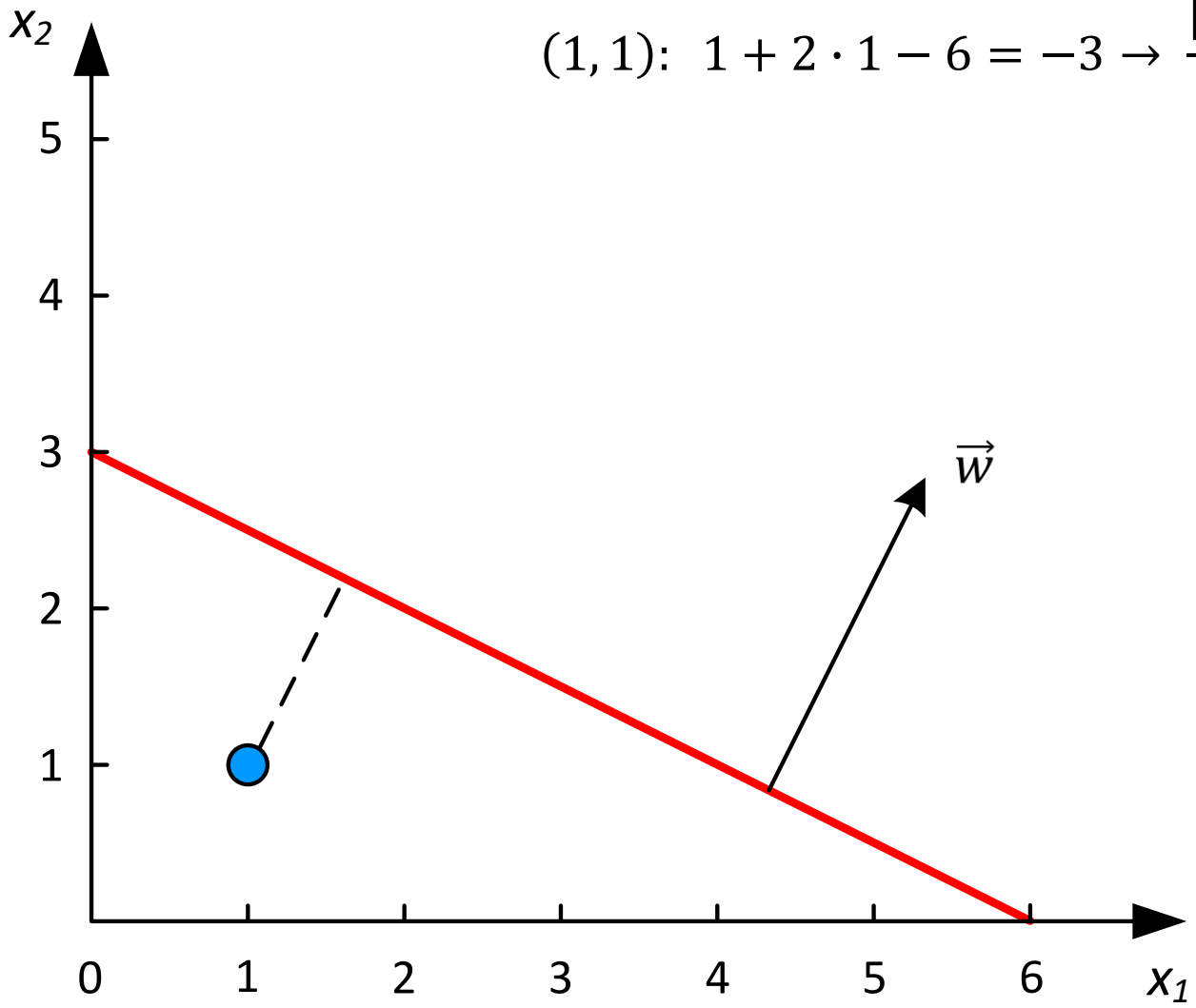
$$\vec{w} = (1, 2), \qquad \|\vec{w}\| = \sqrt{1^2 + 2^2} = \sqrt{5} \cong 2.236$$

Геометрическая интерпретация



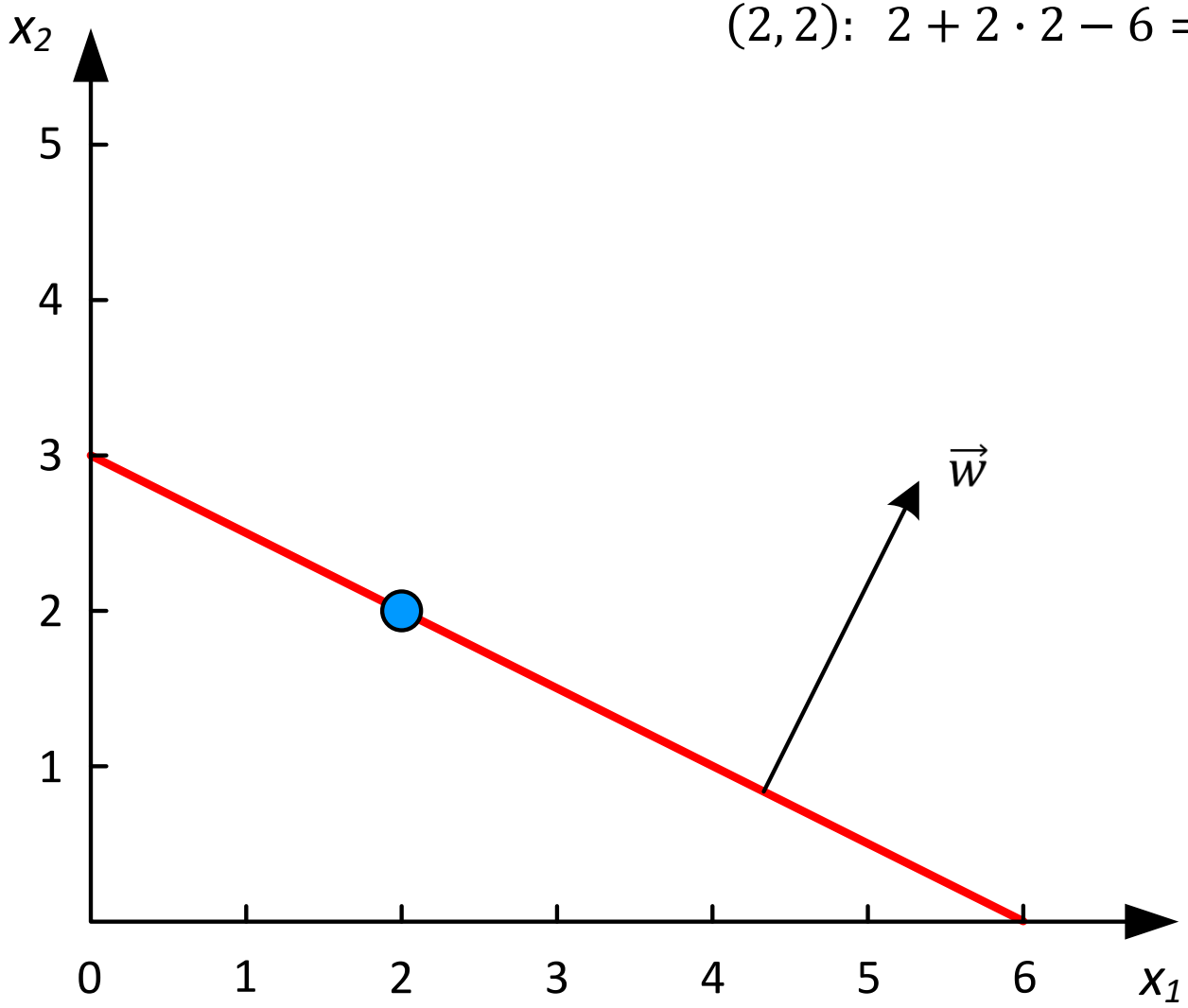
Геометрическая интерпретация

$$(1,1):\ 1+2\cdot 1-6=-3\rightarrow \frac{|-3|}{\sqrt{5}}\cong 1.34$$



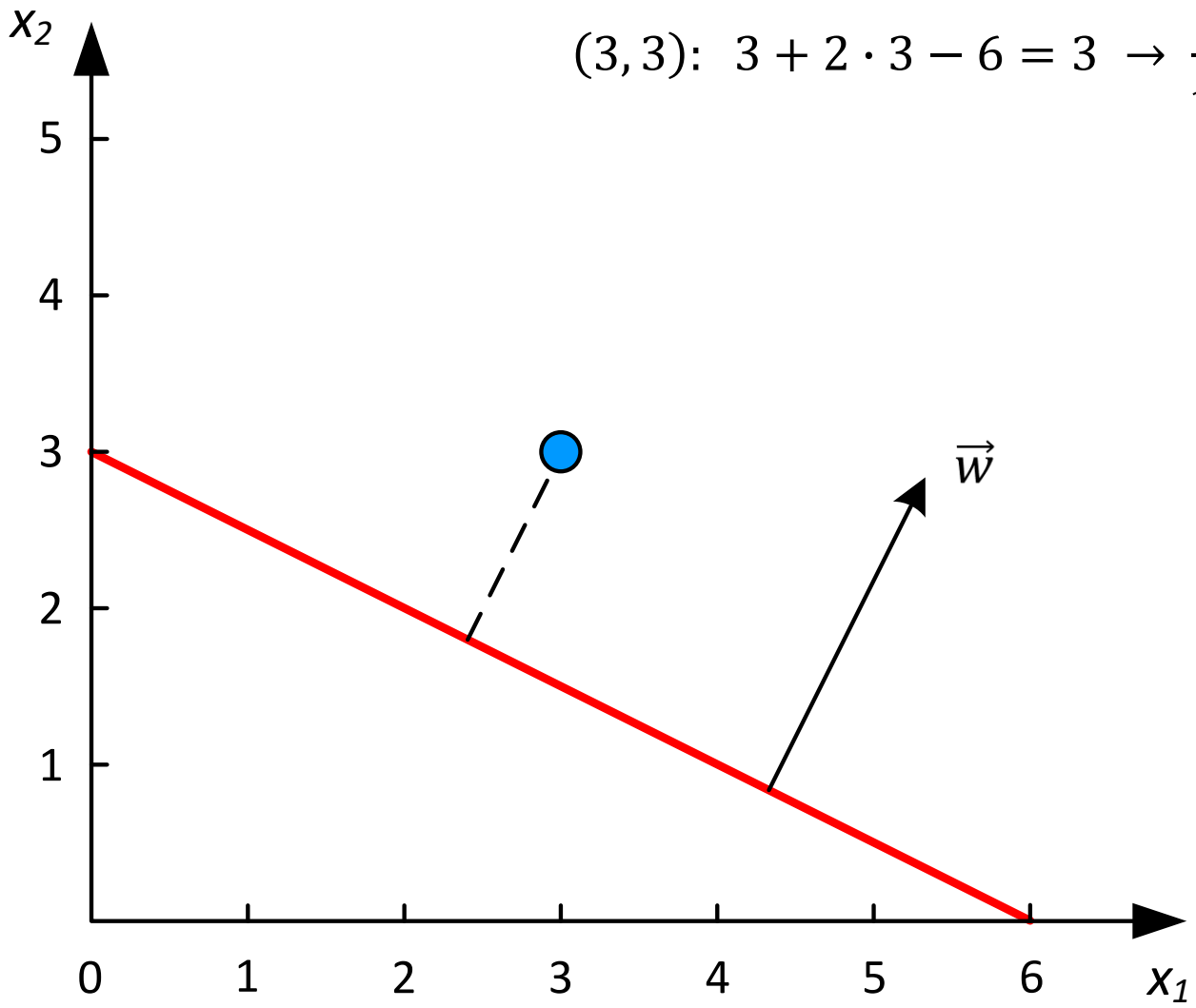
Геометрическая интерпретация

$(2, 2)$: $2 + 2 \cdot 2 - 6 = 0$



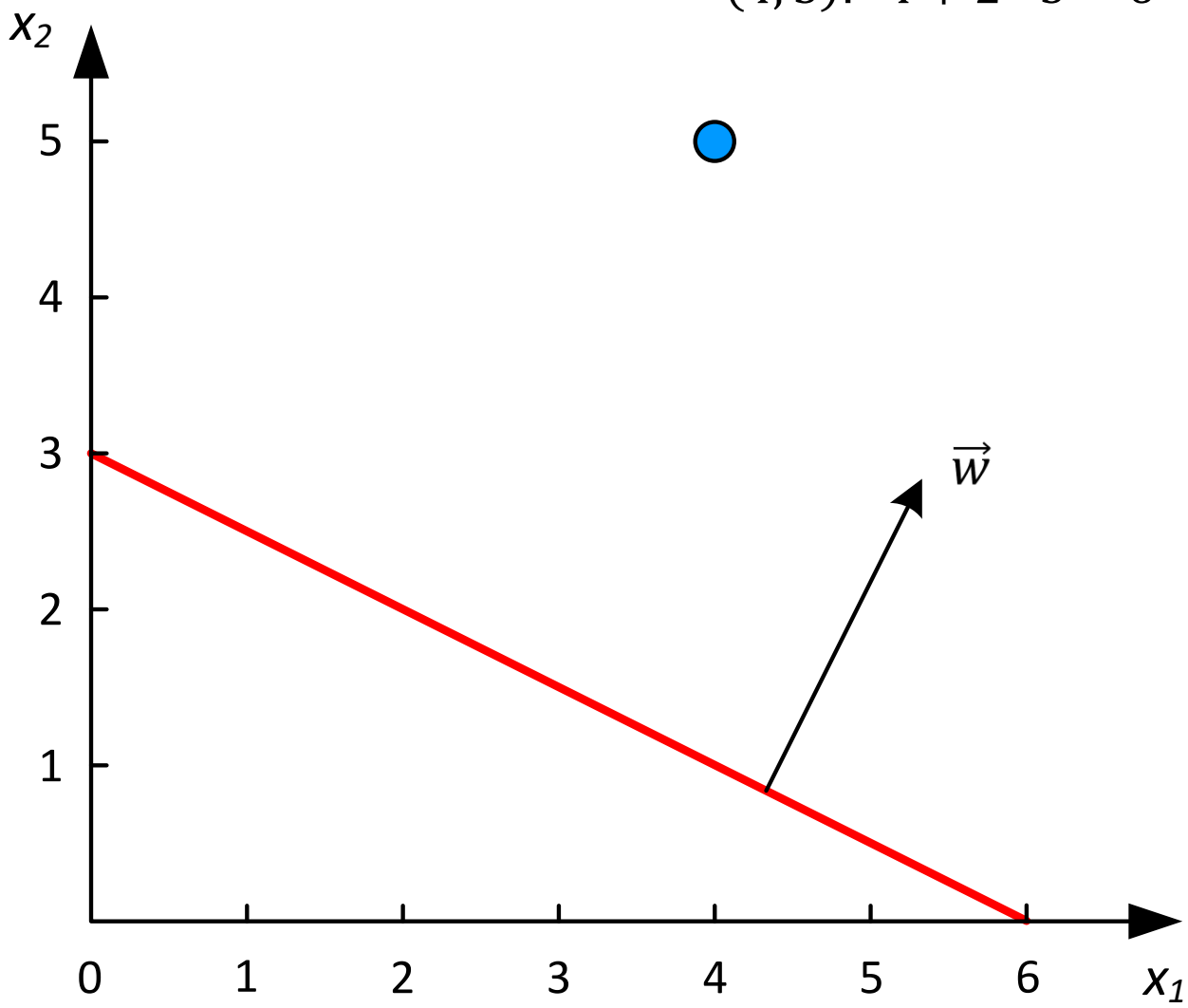
Геометрическая интерпретация

$$(3,3):\ 3 + 2 \cdot 3 - 6 = 3 \ \rightarrow \ \frac{|3|}{\sqrt{5}} \cong 1.34$$



Геометрическая интерпретация

$$(4,5):\ 4 + 2 \cdot 5 - 6 = 8 \ \rightarrow \ \frac{8}{\sqrt{5}} \cong 3.58$$



Функционалы качества/ошибки

- Доля правильных ответов или *правильность* (accuracy):

$$Q(a, X) = \frac{1}{l}\sum_{i=1}^{l}[a(\vec{x}_i) = y_i],$$

где $[\cdot]$ – нотация (скобка) Айверсона:

$$[P] = \begin{cases} 1, \text{если } P - \text{истинно} \\ 0, \text{если } P - \text{ложно} \end{cases}$$

- Доля неправильных ответов:

$$Q(a, X) = \frac{1}{l}\sum_{i=1}^{l}[a(\vec{x}_i) \neq y_i] = \frac{1}{l}\sum_{i=1}^{l}[sign\langle\vec{w}, \vec{x}_i\rangle \neq y_i] \to \min_{\vec{w}}$$

– дискретная функция

Функционал ошибки

- Модифицированный вариант:

$$Q(a, X) = \frac{1}{l} \sum_{i=1}^{l} [y_i \langle \vec{w}, \vec{x}_i \rangle < 0] \to \min_{\vec{w}}$$

- Величина $M = y_i \langle \vec{w}, \vec{x}_i \rangle$ называется *отступом* (margin)
- Знак отступа говорит о корректности ответа классификатора:
  - положительный отступ $\to$ правильный ответ
  - отрицательный отступ $\to$ неправильный ответ
- Абсолютная величина отступа характеризует степень уверенности классификатора в своём ответе

Функция потерь

- Функция потерь (пороговая):

$$L(M) = [M < 0]$$

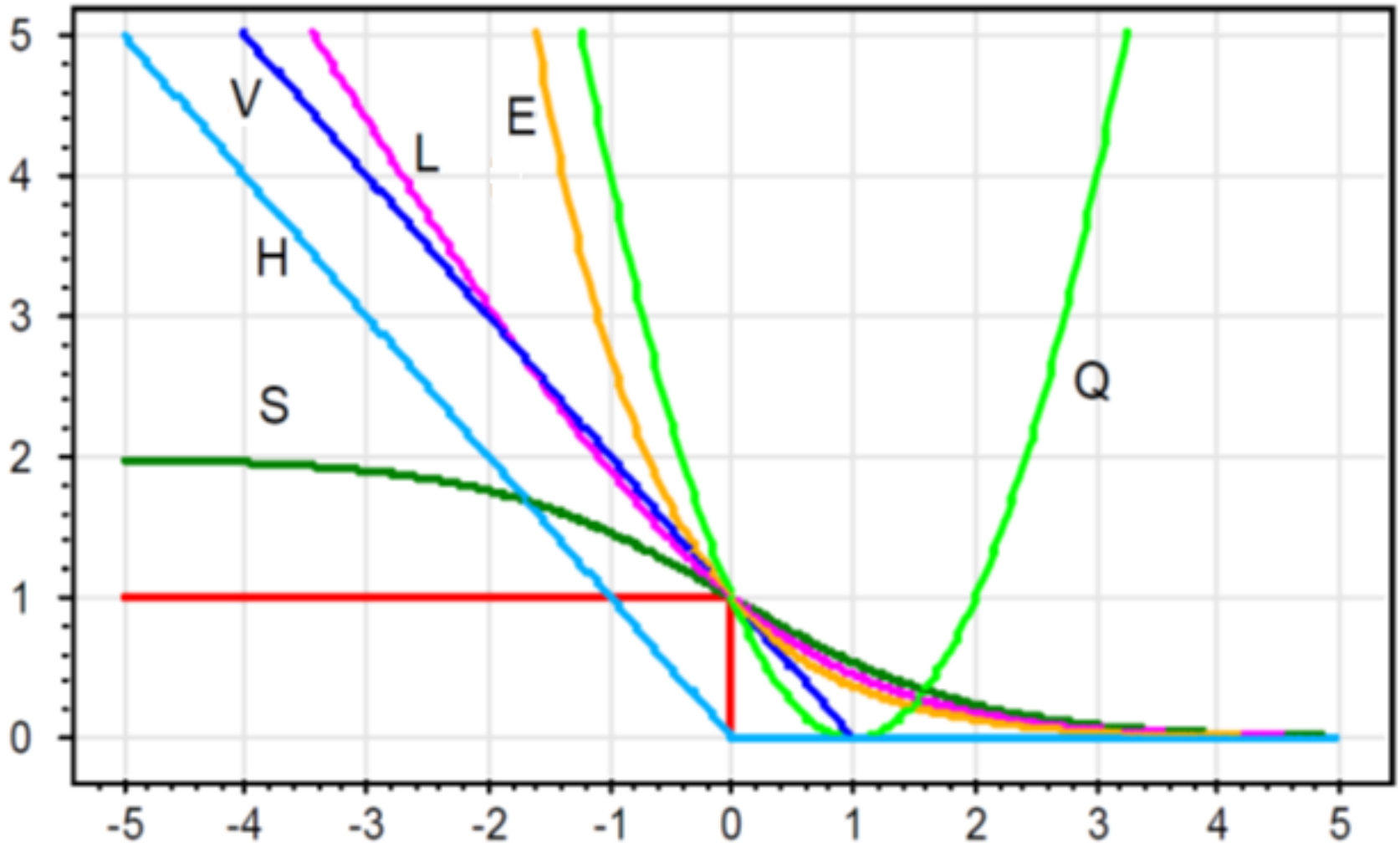
- Если оценить эту функцию сверху:

$$L(M) \leq \tilde{L}(M),$$

то можно получить верхнюю оценку для функционала ошибки:

$$Q(a, X) \leq \frac{1}{l} \sum_{i=1}^{l} \tilde{L}(y_i \langle \vec{w}, \vec{x}_i \rangle) \to \min_{\vec{w}}$$

Функция потерь



Функция потерь

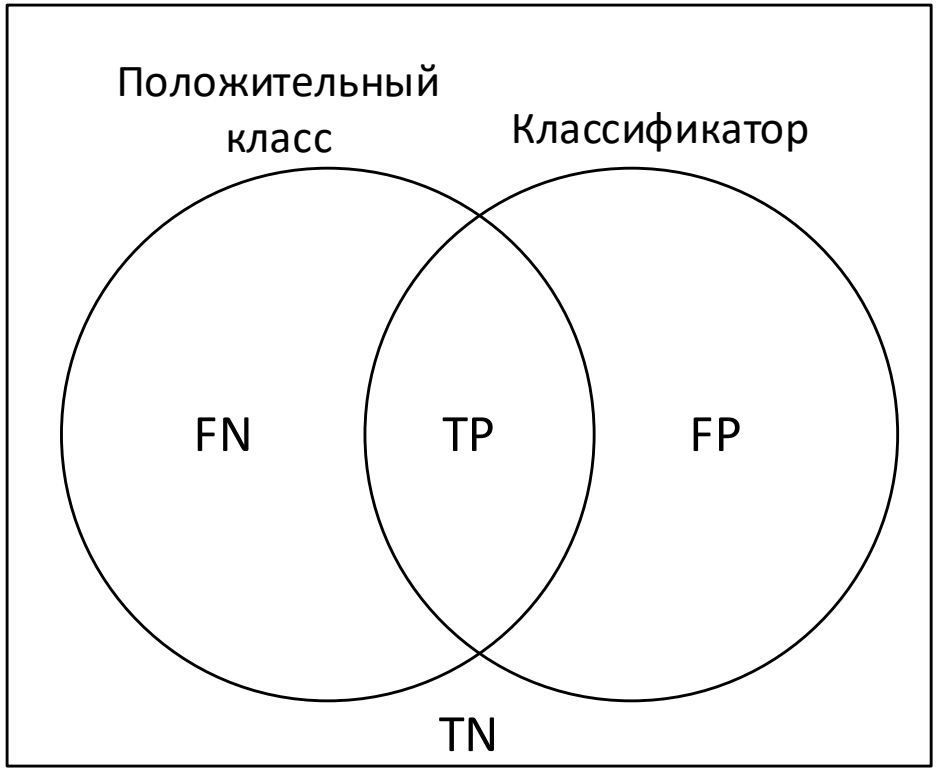
1. $\tilde{L}(M) = log(1 + e^{-M})$ – логистическая функция потерь (L)
2. $\tilde{L}(M) = (1 - M)_+ = max(0,1 - M)$ – кусочно-линейная функция потерь (hinge loss) (метод опорных векторов) (V)
3. $\tilde{L}(M) = (-M)_+ = max(0, -M)$ – кусочно-линейная функция потерь (персептрон Розенблатта) (H)
4. $\tilde{L}(M) = e^{-M}$ – экспоненциальная функция потерь (AdaBoost) (E)
5. $\tilde{L}(M) = \frac{2}{(1+e^{M})}$ – сигмоидная функция потерь (S)
6. $\tilde{L}(M) = (1 - M)^2$ – квадратичная функция потерь (Q)

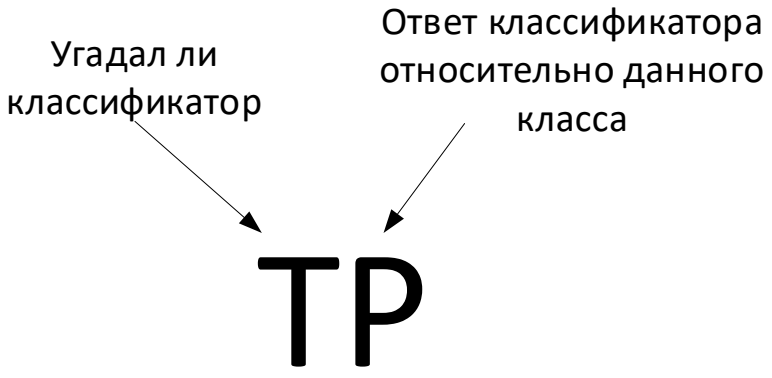
Метрики качества классификации

Матрица ошибок (confusion matrix):

		Оценка классификатора	
		$a(x) = +1$ (Positive)	$a(x) = -1$ (Negative)
Истинные ответы	$y = +1$	TP (True Positive)	FN (False Negative)
	$y = -1$	FP (False Positive)	TN (True Negative)

Метрики качества классификации





Метрики качества классификации

- Accuracy (Правильность):

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN}$$

- Precision (Точность):

$$P = \frac{TP}{TP + FP}$$

- Recall (Полнота):

$$R = \frac{TP}{TP + FN}$$

- F1-measure (F1-score, F1-мера):

$$F1 = \frac{2PR}{P + R}$$

Метрики качества классификации

- Микроусреднение (micro-averaging):

$$P^{micro} = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n}(TP_i + FP_i)}, \qquad R^{micro} = \frac{\sum_{i=1}^{n} TP_i}{\sum_{i=1}^{n}(TP_i + FN_i)},$$

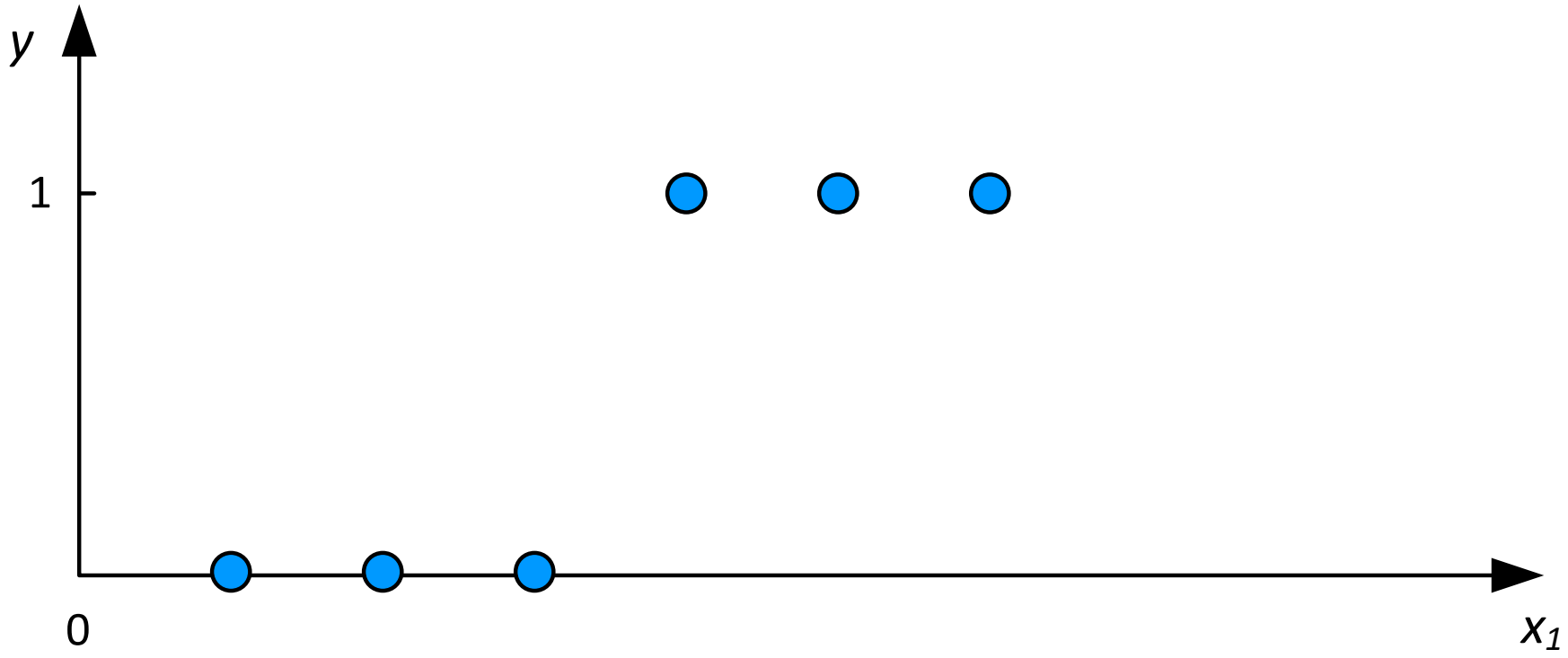
$$F_1^{micro} = \frac{2P^{micro}R^{micro}}{P^{micro} + R^{micro}}$$

- Макроусреднение (macro-averaging):

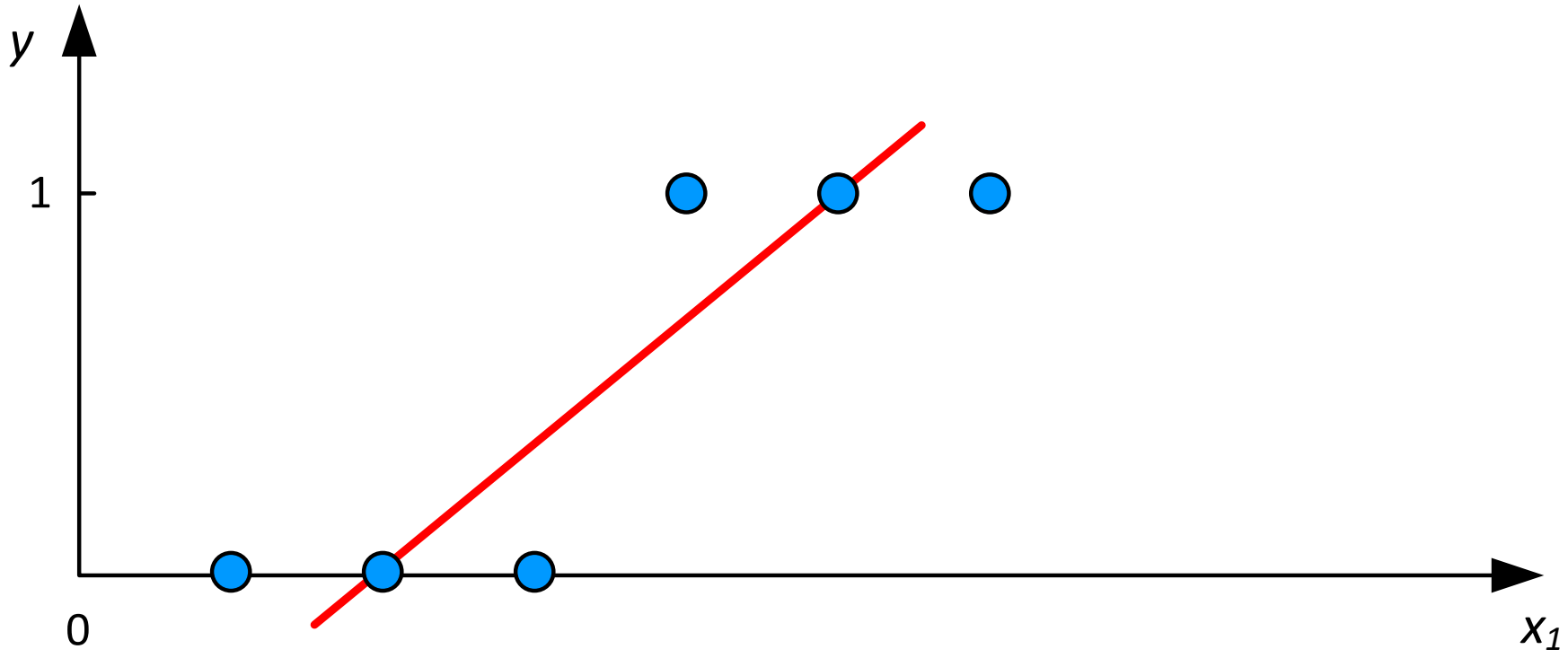
$$P^{macro} = \frac{\sum_{i=1}^{n} P_i}{n}, \qquad R^{macro} = \frac{\sum_{i=1}^{n} R_i}{n},$$

$$F_1^{macro} = \frac{\sum_{i=1}^{n} F_{1i}}{n}$$

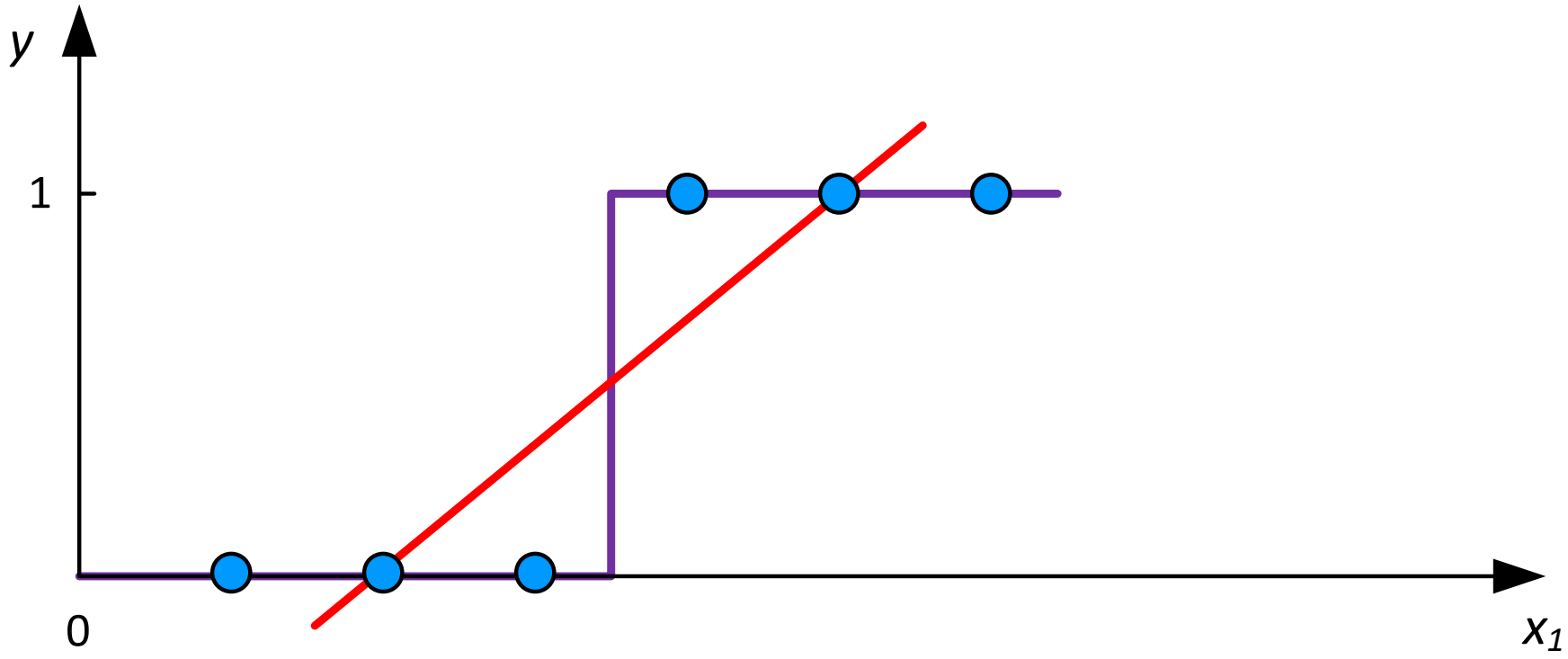
Логистическая регрессия



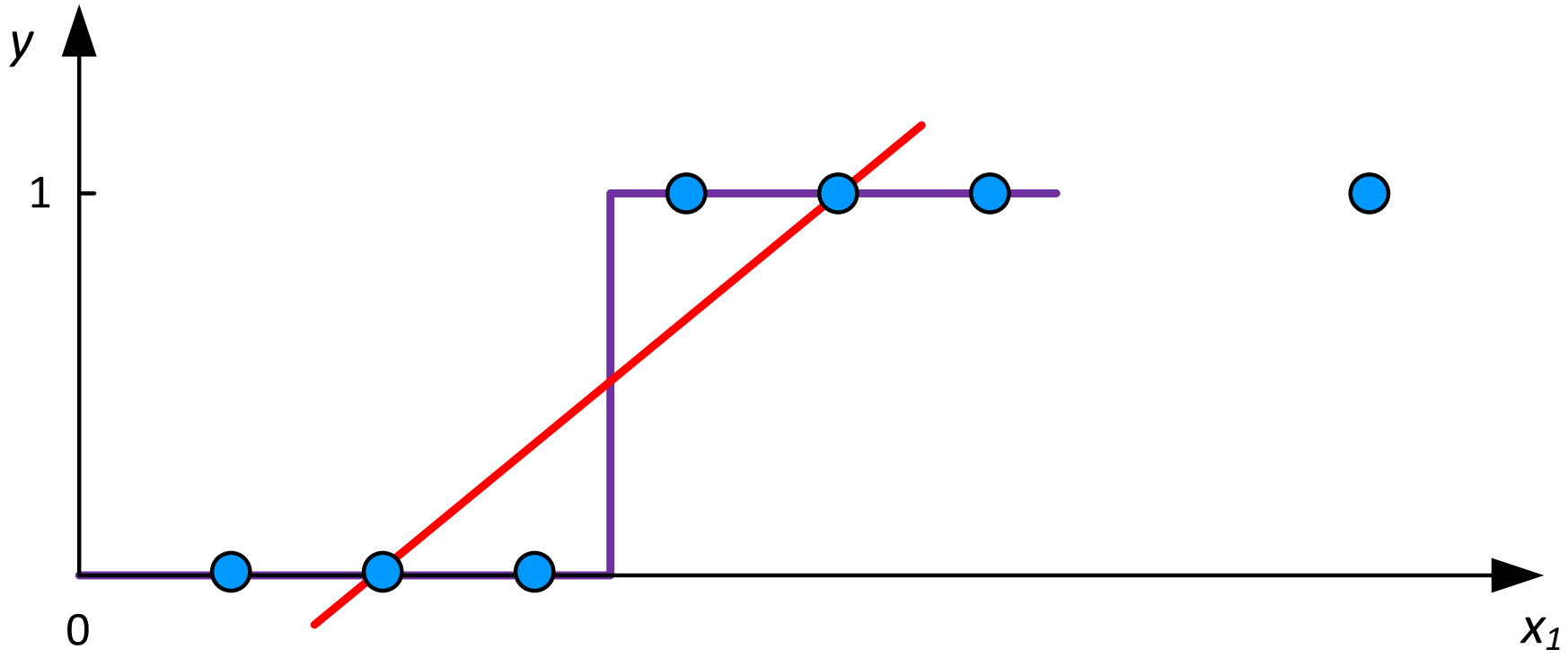
Логистическая регрессия



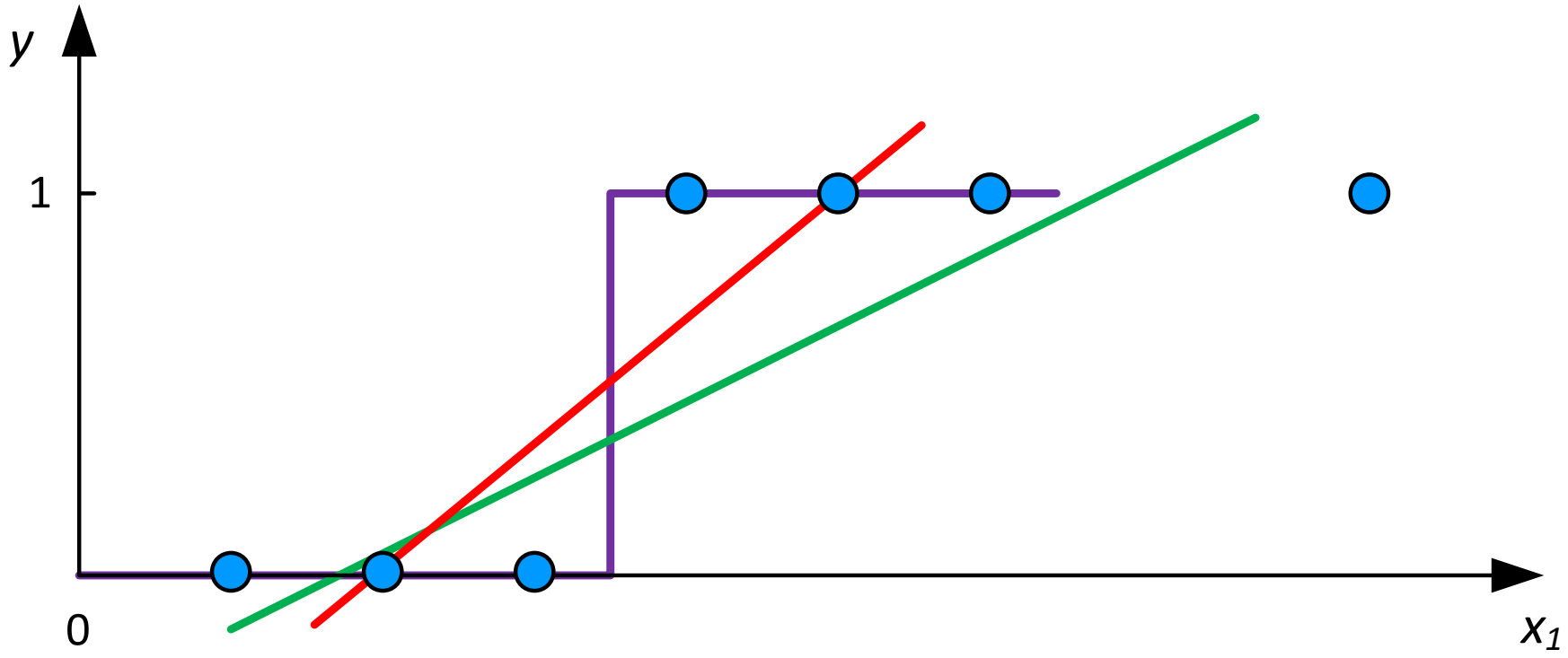
Логистическая регрессия



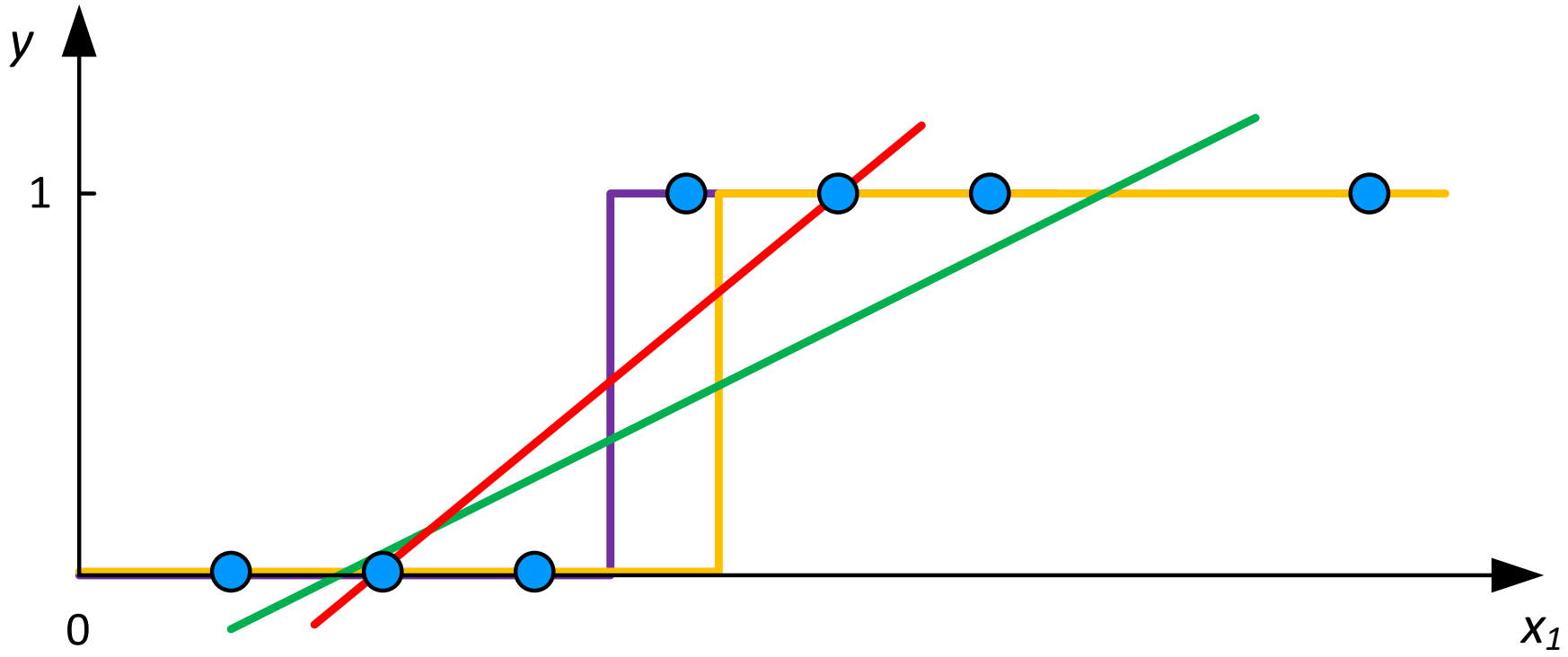
Логистическая регрессия



Логистическая регрессия



Логистическая регрессия



Логистическая регрессия

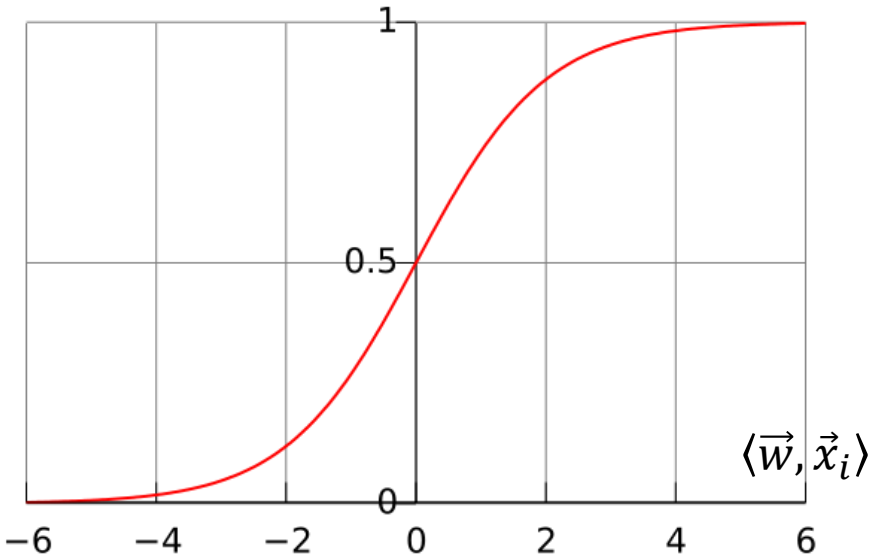
- Линейная регрессия:

$$a(\vec{x}_i) = \langle \vec{w}, \vec{x}_i \rangle$$

- Логистическая регрессия:

$$a(\vec{x}_i) = g(\langle \vec{w}, \vec{x}_i \rangle) = \frac{1}{1 + e^{-\langle \vec{w}, \vec{x}_i \rangle}}$$

- Сигмоидальная (логистическая) функция:



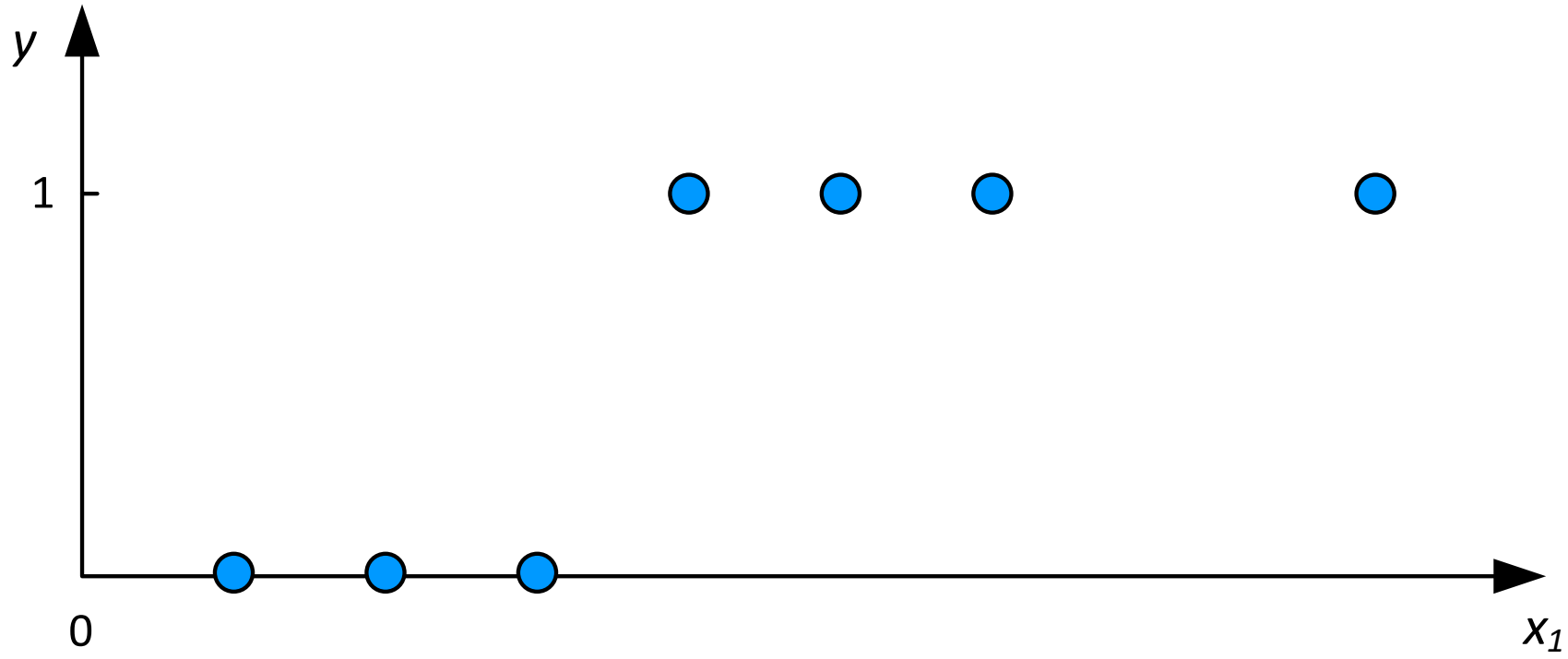
Логистическая регрессия

- Пусть $y \in \{0,1\}$
- Тогда:
  - $a(\vec{x}_i)$ – вероятность того, что объект $\vec{x}_i$ принадлежит классу $y_i = 1$
  - $1 - a(\vec{x}_i)$ – вероятность того, что объект $\vec{x}_i$ принадлежит классу $y_i = 0$
- Интерпретация выходных значений:

$$y = \begin{cases} 1, \text{если } a(\vec{x}_i) \geq 0.5, \text{т. е.} \langle \vec{w}, \vec{x}_i \rangle \geq 0 \\ 0, \text{если } a(\vec{x}_i) < 0.5, \text{т. е.} \langle \vec{w}, \vec{x}_i \rangle < 0 \end{cases}$$

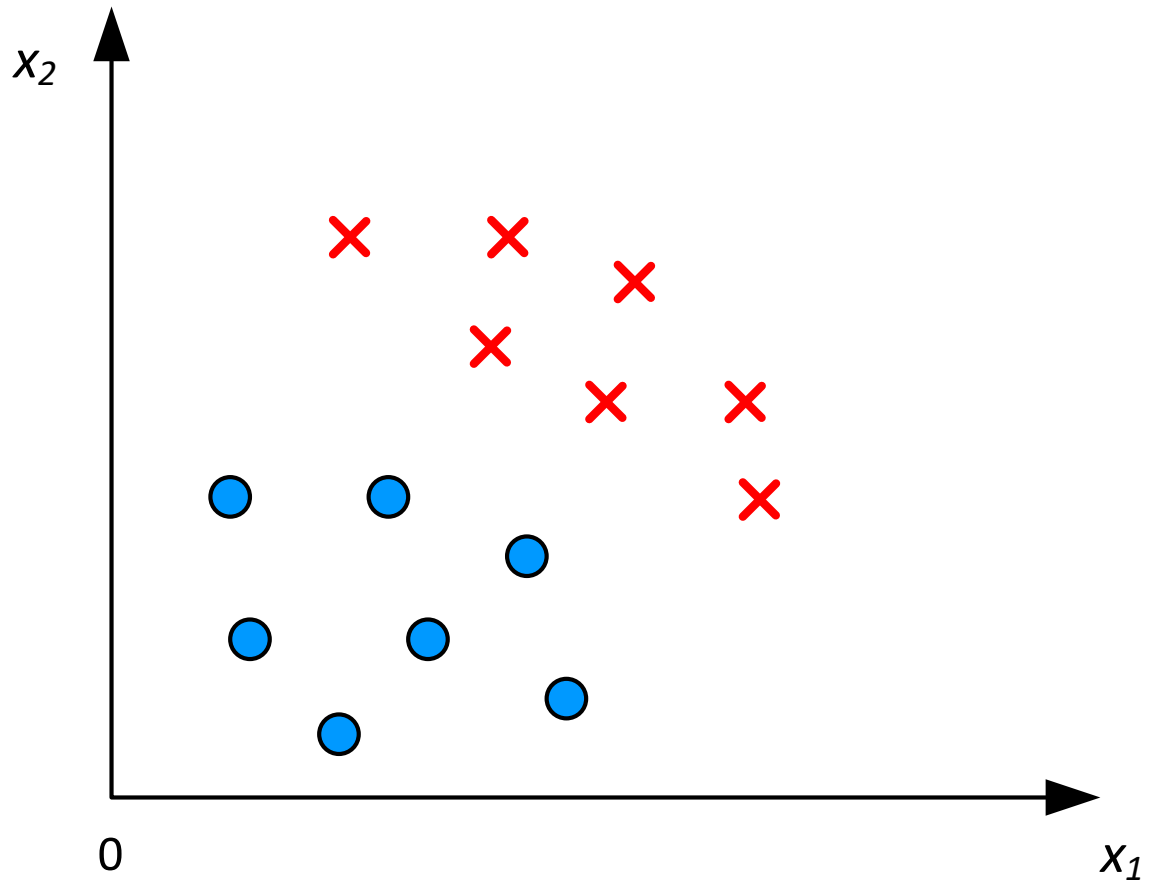
Геометрическая интерпретация

- Одномерный случай:



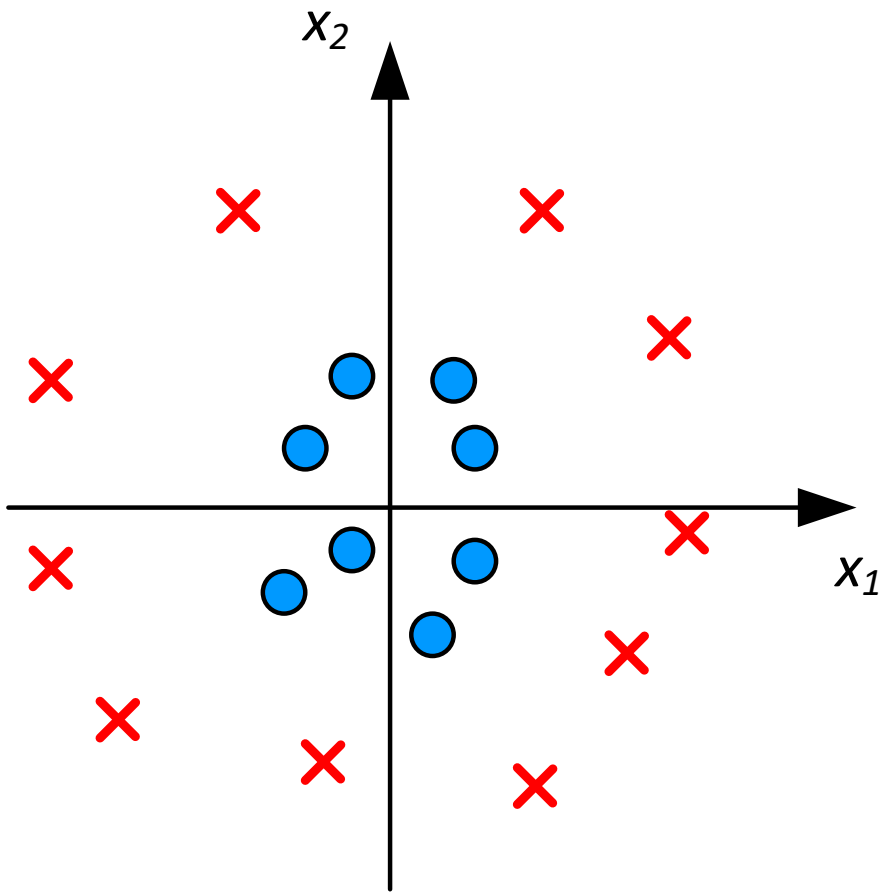
Геометрическая интерпретация

- Двумерный случай:



Геометрическая интерпретация

- Двумерный нелинейный случай:



$$a(\vec{x}) =$$

$$= g(w_0 + w_1 x_1 + w_2 x_2 +$$

$$+ w_3 x_1^2 + w_4 x_2^2)$$

Обучение логистической регрессии

- Условные вероятности:

$$P(y = 1|x) = a(x)$$

$$P(y = 0|x) = 1 - a(x)$$

- В компактном виде:

$$P(y|x) = \big(a(x)\big)^y \big(1 - a(x)\big)^{1-y}$$

- *Правдоподобие выборки* (likelihood) – вероятность получить данную выборку с точки зрения алгоритма:

$$P(\vec{y}|X) = \prod_{i=1}^{l} P(y_i|\vec{x}_i) = \prod_{i=1}^{l} \big(a(\vec{x}_i)\big)^{y_i} \big(1 - a(\vec{x}_i)\big)^{1-y_i}$$

Обучение логистической регрессии

- Правдоподобие:

$$P(\vec{y}|X) = \prod_{i=1}^{l} \left(a(\vec{x}_i)\right)^{y_i} \left(1 - a(\vec{x}_i)\right)^{1-y_i}$$

- *Логарифмическое правдоподобие* (log likelihood):

$$\log P(\vec{y}|X) = \sum_{i=1}^{l} y_i \log a(\vec{x}_i) + (1 - y_i) \log\left(1 - a(\vec{x}_i)\right) \to \max_{\vec{w}}$$

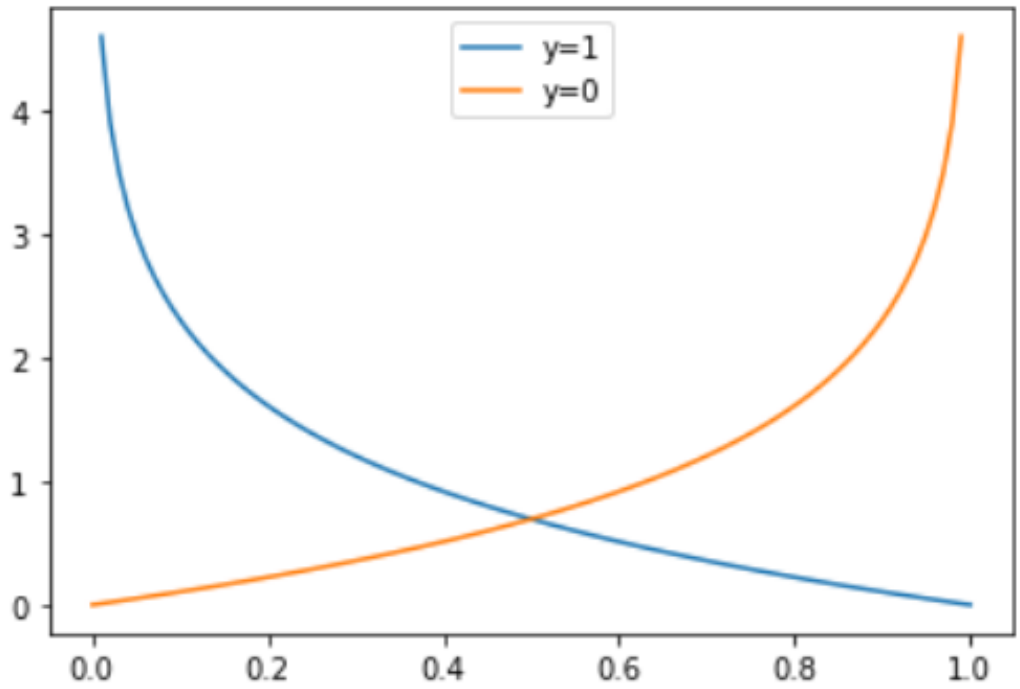
- Переходим к минимизации (log-loss или cross-entropy – *перекрестная энтропия*):

$$-\sum_{i=1}^{l} y_i \log a(\vec{x}_i) + (1 - y_i) \log\left(1 - a(\vec{x}_i)\right) \to \min_{\vec{w}}$$

Обучение логистической регрессии

- На одном объекте (функция потерь):

$$L(a, \vec{x}_i) = \begin{cases} -\log a(\vec{x}_i), y_i = 1 \\ -\log\left(1 - a(\vec{x}_i)\right), y_i = 0 \end{cases}$$



Обучение логистической регрессии

- Функция потерь:

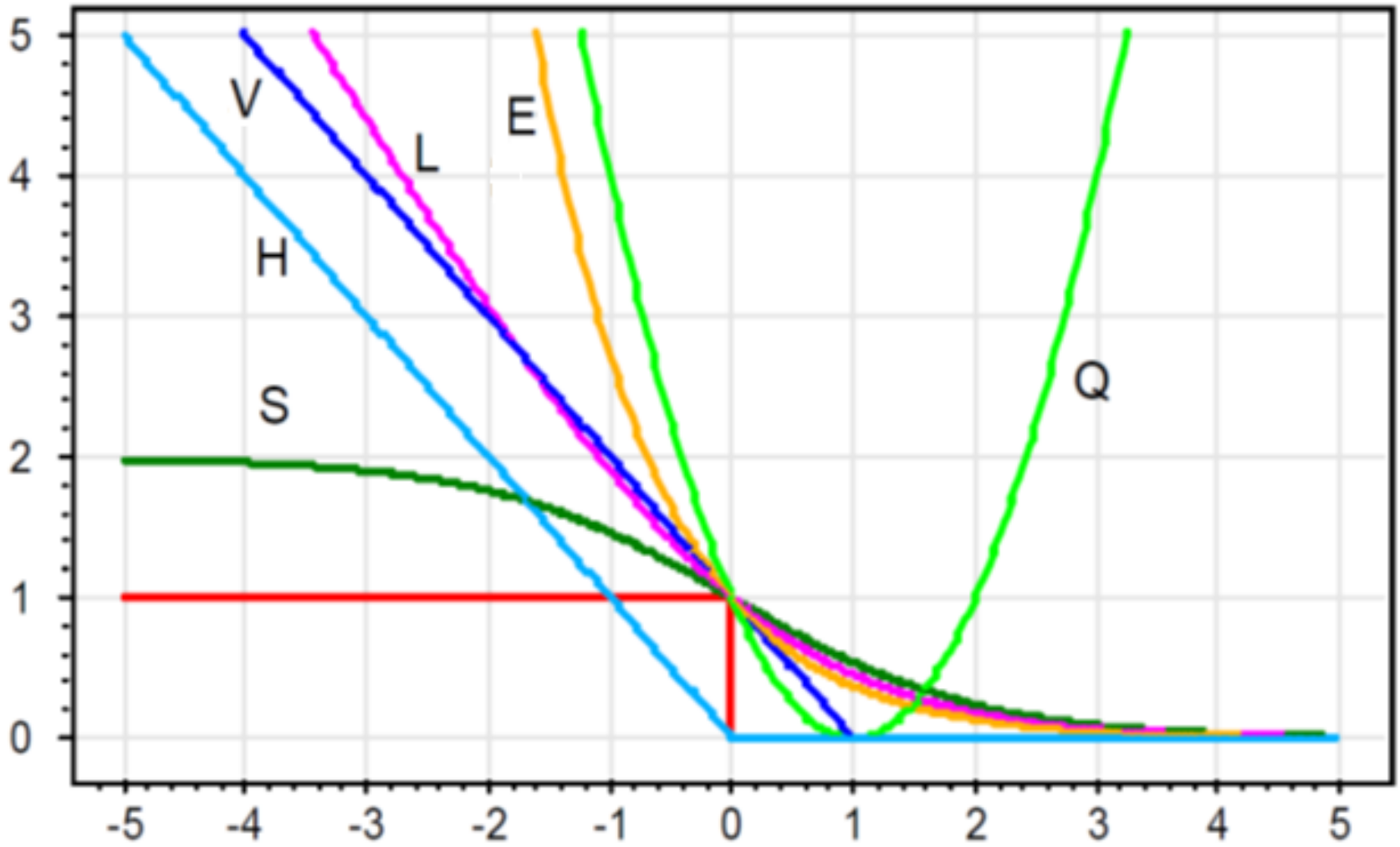
$$L(a, \vec{x}_i) = \begin{cases} -\log a(\vec{x}_i), y_i = 1 \\ -\log\left(1 - a(\vec{x}_i)\right), y_i = -1 \end{cases}$$

$$L(a, \vec{x}_i) = \begin{cases} -\log \dfrac{1}{1 + e^{-\langle \vec{w}, \vec{x}_i \rangle}}, y_i = 1 \\ -\log\left(1 - \dfrac{1}{1 + e^{-\langle \vec{w}, \vec{x}_i \rangle}}\right) = -\log \dfrac{1}{1 + e^{\langle \vec{w}, \vec{x}_i \rangle}}, y_i = -1 \end{cases}$$

$$L(a, \vec{x}_i) = -\log \frac{1}{1 + e^{-y_i \langle \vec{w}, \vec{x}_i \rangle}} = -\log \frac{1}{1 + e^{-M}} = -\log 1 + \log(1 + e^{-M})$$

$$L(M) = \log(1 + e^{-M})$$

Функция потерь



Обучение логистической регрессии

- Градиентный спуск:

$$a'(x) = a(x)\big(1 - a(x)\big)$$

$$\nabla \log P(\vec{y}|X) = \big(\vec{y} - a(X)\big)X$$

$$\frac{\partial}{\partial w_j}(\log P(\vec{y}|X)) = \sum_{i=1}^{l}\big(y_i - a(\vec{x}_i)\big)\,x_{ij}$$

- Шаг градиентного спуска:

$$w_j^{(k)} = w_j^{(k-1)} - \eta_k \sum_{i=1}^{l}\big(y_i - a(\vec{x}_i)\big)\,x_{ij}$$

Метод опорных векторов (Support Vector Machine, SVM)

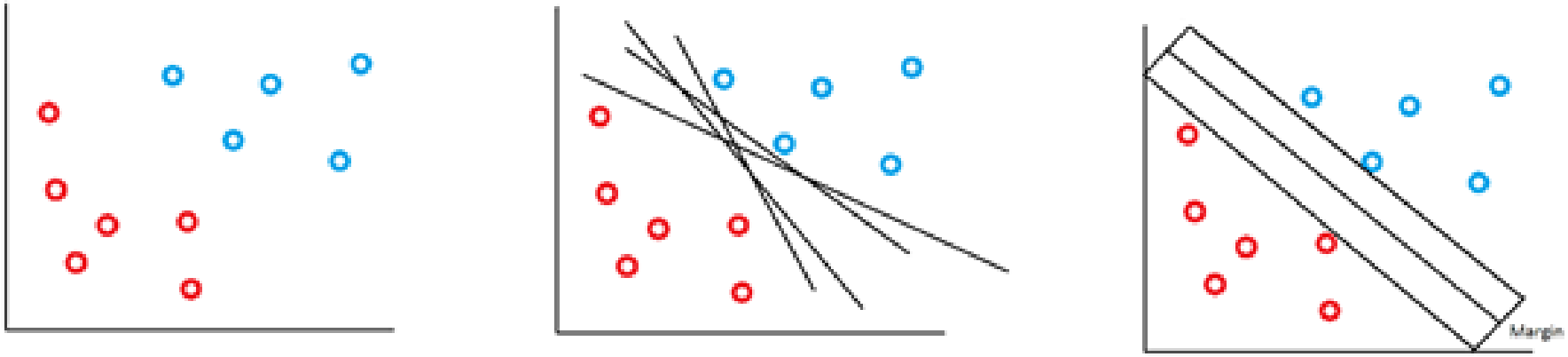
Случай линейной разделимости

- Линейный классификатор:

$$a(\vec{x}) = sign(\langle \vec{w}, \vec{x} \rangle + w_0)$$

- Пусть существуют такие параметры $\vec{w}$ и w_0, что классификатор не допускает ни одной ошибки на обучающей выборке
- В этом случае говорят, что выборка *линейно разделима*

Метод опорных векторов: идея



- Пусть разделяющая гиперплоскость расположена так, что ближайшие объекты обоих классов находятся от неё на одинаковом расстоянии
- Таких разделяющих полос может быть бесконечно много
- В методе опорных векторов ширина разделяющей полосы максимизируется

Метод опорных векторов

- Составим оптимизационную задачу
- Расстояние от произвольной точки $\vec{x}_k$ до гиперплоскости:

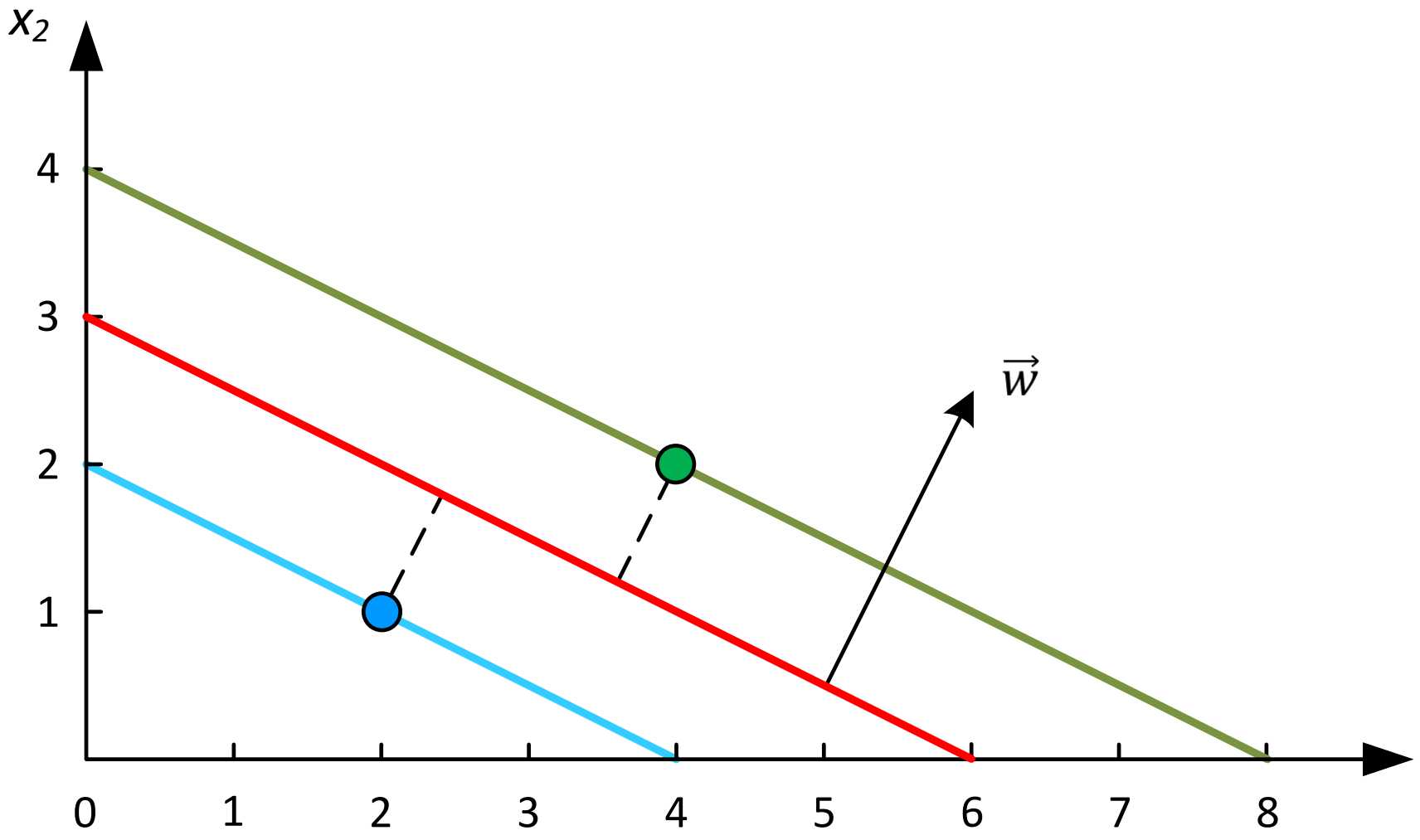
$$d(\vec{x}_k) = \frac{|\langle \vec{w}, \vec{x}_k \rangle + w_0|}{\|\vec{w}\|}$$

- Известно, что если умножить одновременно параметры $\vec{w}$ и w_0 на одну и ту же константу, то классификатор не изменится
- Подберем эту константу так, чтобы

$$\min_{\vec{x}_k \in X} |\langle \vec{w}, \vec{x}_k \rangle + w_0| = 1$$

- Тогда расстояние от разделяющей гиперплоскости до ближайшего объекта обучающей выборки (*опорного вектора*): $1/\|\vec{w}\|$
  - данная величина называется *отступом* (margin)
- Ширина разделяющей полосы равна $2/\|\vec{w}\|$
- Требуется максимизировать ширину разделяющей полосы при условии правильной классификации всех примеров

Метод опорных векторов



Метод опорных векторов

- Уравнения прямых:

$$\left.\begin{array}{l} x_1 + 2x_2 - 8 = 0 \\ x_1 + 2x_2 - 6 = 0 \\ x_1 + 2x_2 - 4 = 0 \end{array}\right\} \quad \vec{w} = (1, 2)$$

- Подберем $\vec{w}$ так, чтобы $|\langle \vec{w}, \vec{x}_0 \rangle + w_0| = 1$:

$$\left.\begin{array}{l} (4,2): 1 \cdot 4 + 2 \cdot 2 - 6 = 2 \mid \times \frac{1}{2} \\ (4,2): \frac{1}{2} \cdot 4 + 1 \cdot 2 - 3 = 1 \\ (2,1): \frac{1}{2} \cdot 2 + 1 \cdot 1 - 3 = -1 \end{array}\right\} \vec{w} = \left(\frac{1}{2}, 1\right)$$

Метод опорных векторов

- Расстояние от точки до плоскости:

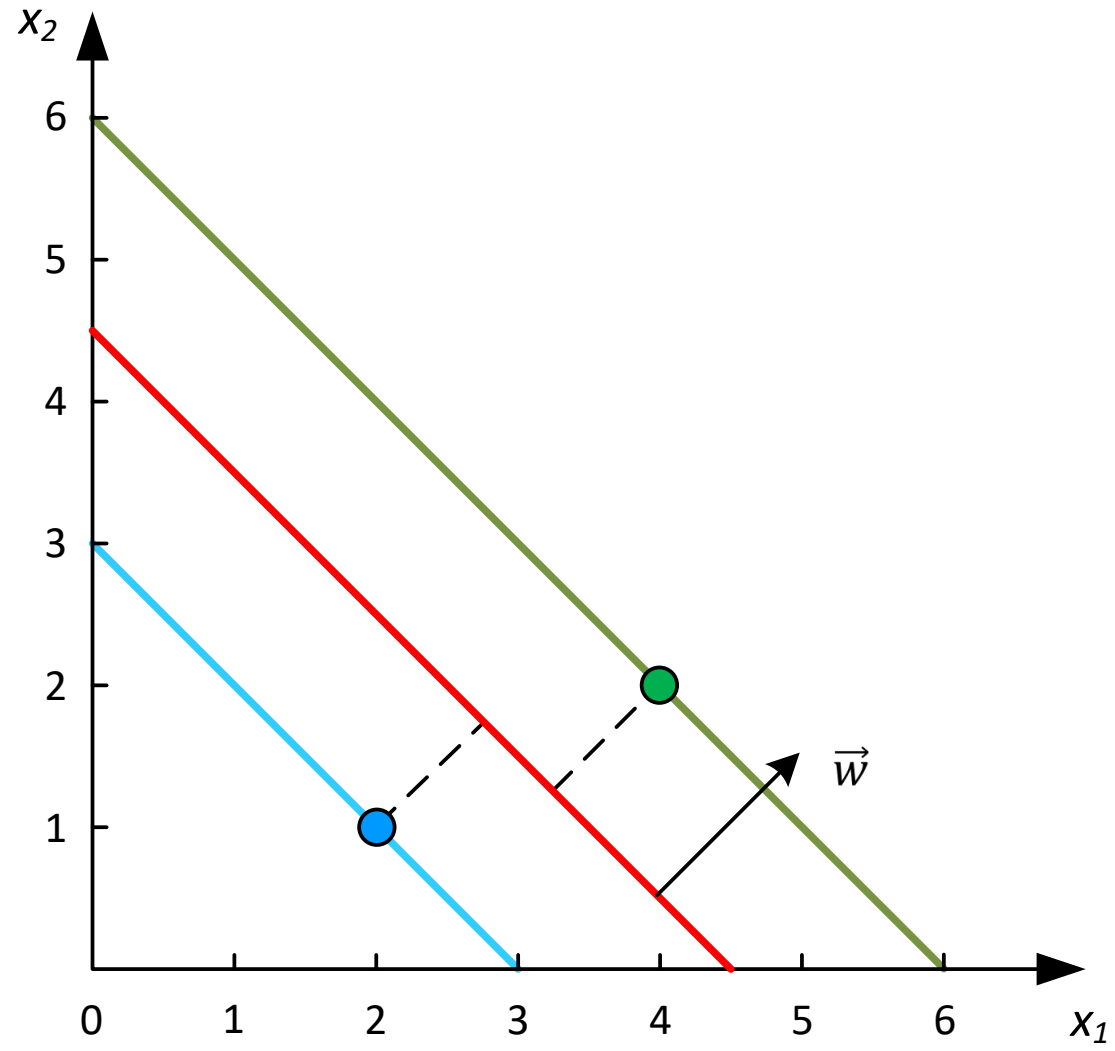
$$d(\vec{x}_k) = \frac{|\langle \vec{w}, \vec{x}_k \rangle + w_0|}{\|\vec{w}\|}$$

$$(4,2): \frac{\left|\frac{1}{2} \cdot 4 + 1 \cdot 2 - 3\right|}{\sqrt{\left(\frac{1}{2}\right)^2 + 1^2}} = \frac{1}{\sqrt{\frac{5}{4}}} \cong 0{,}894$$

- Ширина разделяющей полосы:

$$\frac{2}{\|\vec{w}\|} = \frac{2}{\sqrt{\frac{5}{4}}} \cong 1{,}789$$

Метод опорных векторов



Метод опорных векторов

- Уравнения прямых:

$$\left.\begin{array}{c} x_1 + x_2 - 6 = 0 \\ x_1 + x_2 - 4{,}5 = 0 \\ x_1 + x_2 - 3 = 0 \end{array}\right\} \quad \vec{w} = (1, 1)$$

- Подберем $\vec{w}$ так, чтобы $|\langle \vec{w}, \vec{x}_k \rangle + w_0| = 1$:

$$\left.\begin{array}{c} (4,2): 1 \cdot 4 + 1 \cdot 2 - 4{,}5 = 1{,}5 \mid \times \frac{2}{3} \\ (4,2): \frac{2}{3} \cdot 4 + \frac{2}{3} \cdot 2 - 3 = 1 \\ (2,1): \frac{2}{3} \cdot 2 + \frac{2}{3} \cdot 1 - 3 = -1 \end{array}\right\} \vec{w} = \left(\frac{2}{3}, \frac{2}{3}\right)$$

Метод опорных векторов

- Расстояние от точки до плоскости:

$$d(\vec{x}_k) = \frac{|\langle \vec{w}, \vec{x}_k \rangle + w_0|}{\|\vec{w}\|}$$

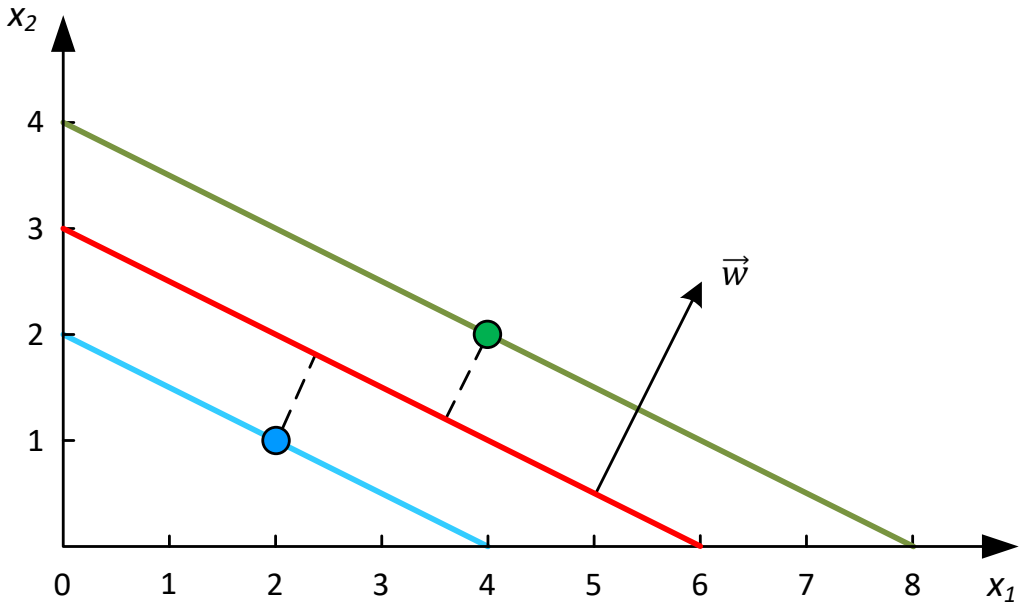
$$(4,2): \frac{\left|\frac{2}{3} \cdot 4 + \frac{2}{3} \cdot 2 - 3\right|}{\sqrt{\left(\frac{2}{3}\right)^2 + \left(\frac{2}{3}\right)^2}} = \frac{1}{\sqrt{\frac{8}{9}}} \cong 1{,}061$$

- Ширина разделяющей полосы:

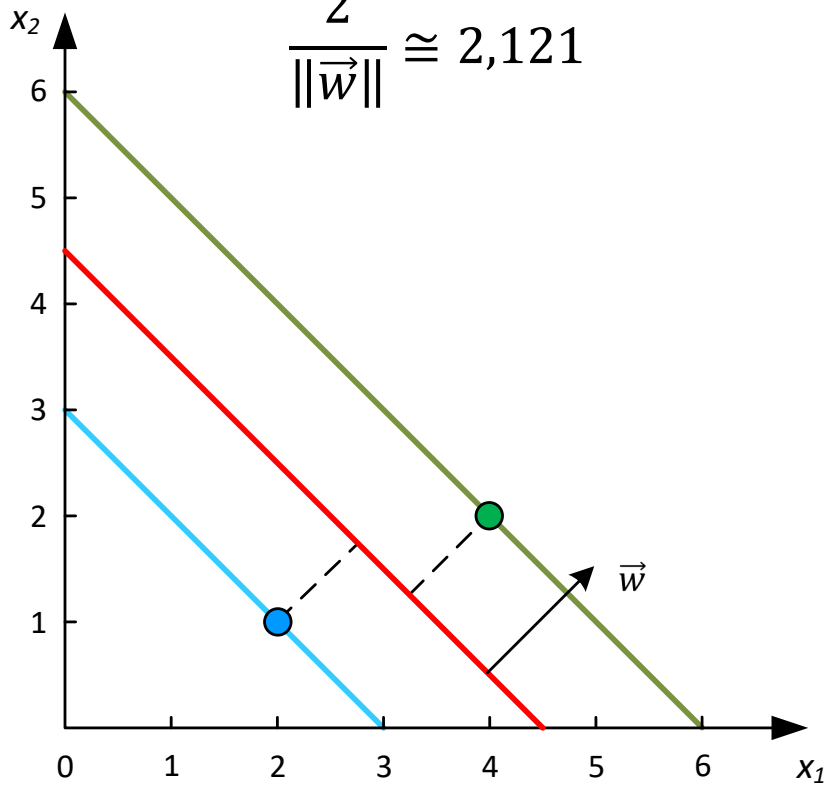
$$\frac{2}{\|\vec{w}\|} = \frac{2}{\sqrt{\frac{8}{9}}} \cong 2{,}121$$

Метод опорных векторов

$$\frac{2}{\|\vec{w}\|} \cong 1{,}789$$



$$\frac{2}{\|\vec{w}\|} \cong 2{,}121$$



Метод опорных векторов

- Оптимизационная задача:

$$\begin{cases} \dfrac{1}{2}\|\vec{w}\|^2 \to \min\limits_{\vec{w}} \\ y_i(\langle \vec{w}, \vec{x}_i \rangle + w_0) \geq 1, \qquad i = 1, \ldots, l \end{cases}$$

- Задача квадратичной оптимизации – найти минимум квадратичной функции при l ограничениях-неравенствах
  - Имеет единственное решение

Метод опорных векторов

Случай отсутствия линейной разделимости

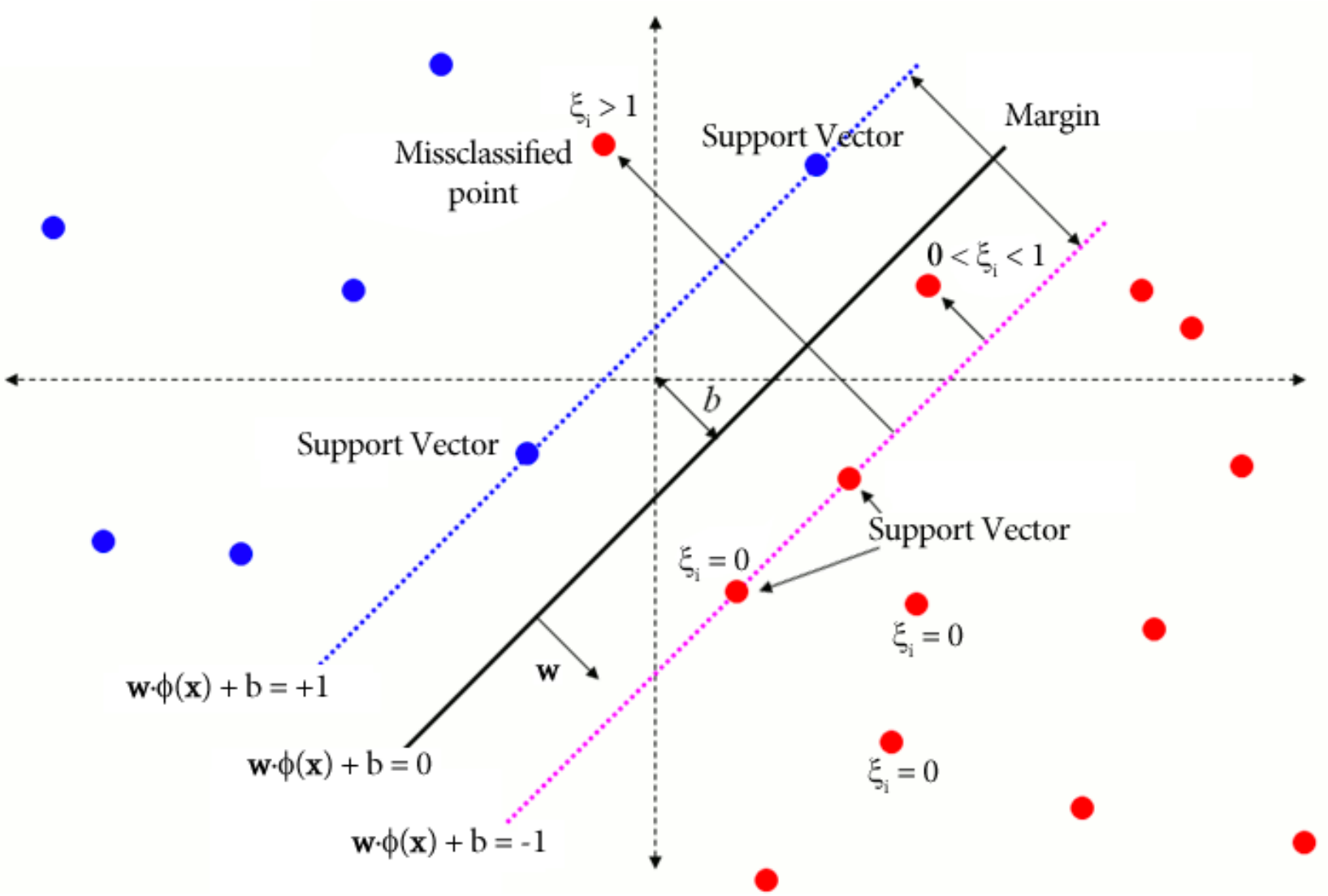
- При любых $\vec{w}$ и w_0 хотя бы одно из ограничений будет нарушено:

$$\exists x_i \in X\colon\ y_i(\langle \vec{w}, \vec{x}_i \rangle + w_0) < 1$$

- Смягчим ограничения, введя штраф $\xi_i \geq 0$ за их нарушение:

$$y_i(\langle \vec{w}, \vec{x}_i \rangle + w_0) \geq 1 - \xi_i$$

Метод опорных векторов



Метод опорных векторов

- В оптимизационной задаче будем максимизировать ширину разделяющей полосы и минимизировать штраф за ошибки:

$$\begin{cases} \dfrac{1}{2}\|\vec{w}\|^2 + C\sum_{i=1}^{l}\xi_i \to \min\limits_{\vec{w},\xi} \\ y_i(\langle \vec{w}, \vec{x}_i\rangle + w_0) \geq 1 - \xi_i, \qquad i = 1, \dots, l \\ \xi_i \geq 0 \end{cases}$$

- C – параметр регуляризации: чем больше параметр C, тем сильнее настраиваемся на обучающую выборку и уменьшаем ширину разделяющей полосы
- Данная задача называется *soft margin SVM* (в отличие от *hard margin*) и также имеет единственное решение
- В процессе решения оптимизационной задачи результат зависит только от попарных скалярных произведений объектов $\langle \vec{x}_i, \vec{x}_j \rangle$

Метод опорных векторов

Функционал ошибки и функция потерь

$$\begin{cases} \frac{1}{2}\|\vec{w}\|^2 + C\sum_{i=1}^{l}\xi_i \to \min\limits_{\vec{w},\xi} \\ y_i(\langle\vec{w},\vec{x}_i\rangle + w_0) \geq 1 - \xi_i, \qquad i = 1,\dots,l \\ \xi_i \geq 0 \end{cases}$$

$$\begin{cases} \xi_i \geq 1 - y_i(\langle\vec{w},\vec{x}_i\rangle + w_0) \\ \xi_i \geq 0 \end{cases}$$

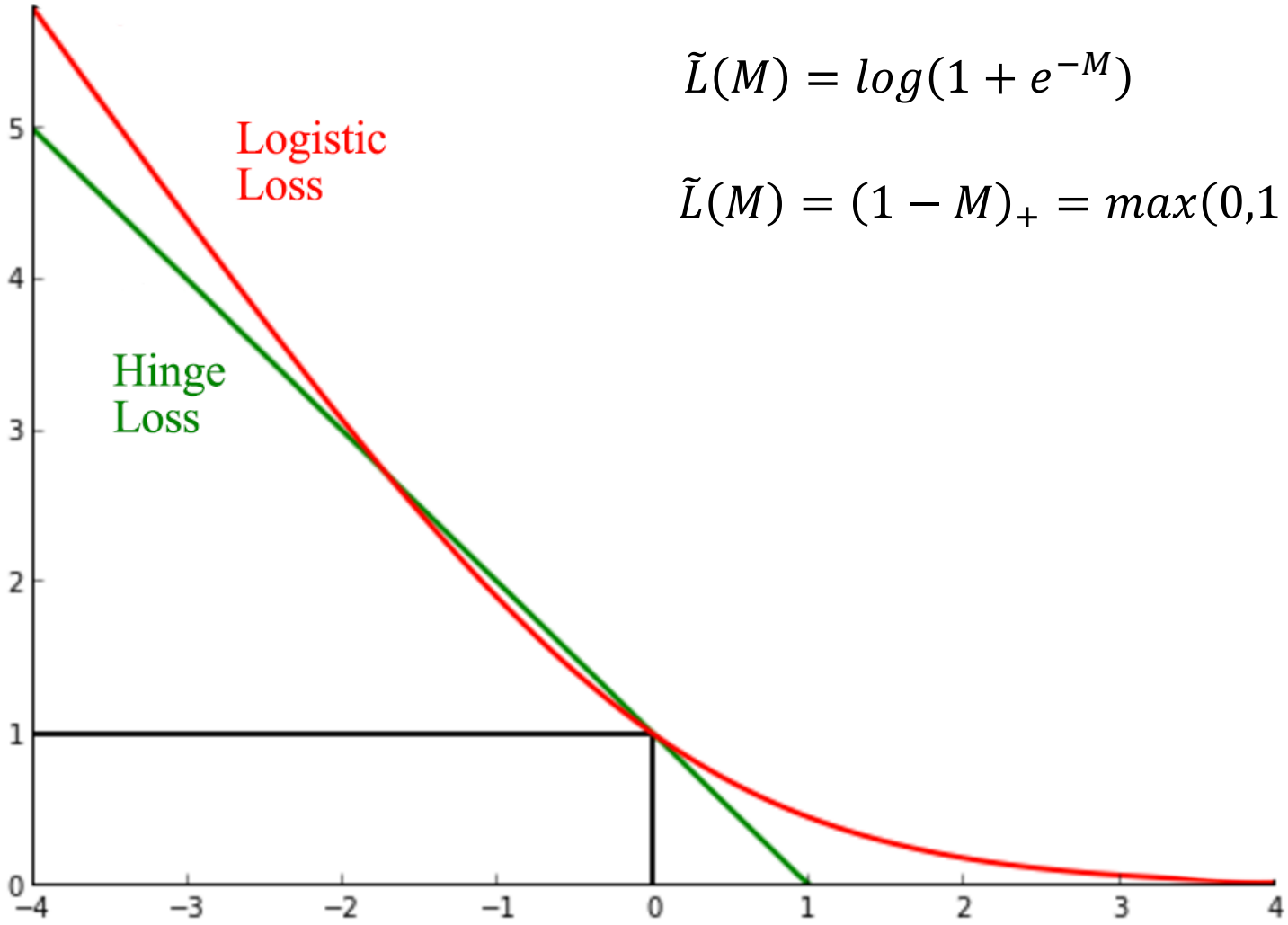
$$\xi_i = \max\big(0, 1 - y_i(\langle\vec{w},\vec{x}_i\rangle + w_0)\big)$$

$$\frac{1}{2}\|\vec{w}\|^2 + C\sum_{i=1}^{l}\max\big(0, 1 - y_i(\langle\vec{w},\vec{x}_i\rangle + w_0)\big) \to \min_{\vec{w},w_0}$$

$$\tilde{L}(M) = (1-M)_+ = max(0,1-M), \qquad \text{где} \quad M = y_i(\langle\vec{w},\vec{x}_i\rangle + w_0)$$

– кусочно-линейная функция потерь (hinge loss)

Метод опорных векторов



$$\tilde{L}(M) = log(1 + e^{-M})$$

$$\tilde{L}(M) = (1 - M)_+ = max(0,1 - M)$$

Метод опорных векторов

Ядра и спрямляющие пространства

- Ещё один подход к решению проблемы линейной неразделимости – переход от исходного пространства признаковых описаний объектов $\mathbb{X}$ к новому пространству $\mathbb{H}$ с помощью некоторого преобразования $\psi: \mathbb{X} \to \mathbb{H}$
- Если пространство $\mathbb{H}$ имеет достаточно высокую размерность, то можно надеяться, что в нём выборка окажется линейно разделимой
- Если выборка X^l не противоречива, то всегда найдётся пространство размерности не более l, в котором она будет линейно разделима
- Пространство $\mathbb{H}$ называют *спрямляющим*.

Метод опорных векторов

- Если предположить, что признаковыми описаниями объектов являются векторы $\vec{\psi}(\vec{x}_i)$, а не векторы $\vec{x}_i$, то построение SVM производится точно так же, как и ранее
- Единственное отличие состоит в том, что скалярное произведение $\langle \vec{x}_i, \vec{x}_j \rangle$ в пространстве $\mathbb{X}$ всюду заменяется скалярным произведением $\langle \vec{\psi}(\vec{x}_i), \vec{\psi}(\vec{x}_j) \rangle$ в пространстве $\mathbb{H}$
- Функция $K(\vec{x}_i, \vec{x}_j) = \langle \vec{\psi}(\vec{x}_i), \vec{\psi}(\vec{x}_j) \rangle$ называется *ядром* (kernel function)
- Алгоритм обучения SVM зависит только от скалярных произведений объектов, но не от самих признаковых описаний
- Поэтому скалярное произведение $\langle \vec{x}_i, \vec{x}_j \rangle$ можно заменить ядром $K(\vec{x}_i, \vec{x}_j)$
- Можно вообще не строить спрямляющее пространство $\mathbb{H}$ в явном виде, и вместо подбора отображения ψ непосредственно подбирать ядра

Метод опорных векторов

Виды ядер:

- линейное: $K(\vec{x}_i, \vec{x}_j) = \langle \vec{x}_i, \vec{x}_j \rangle$

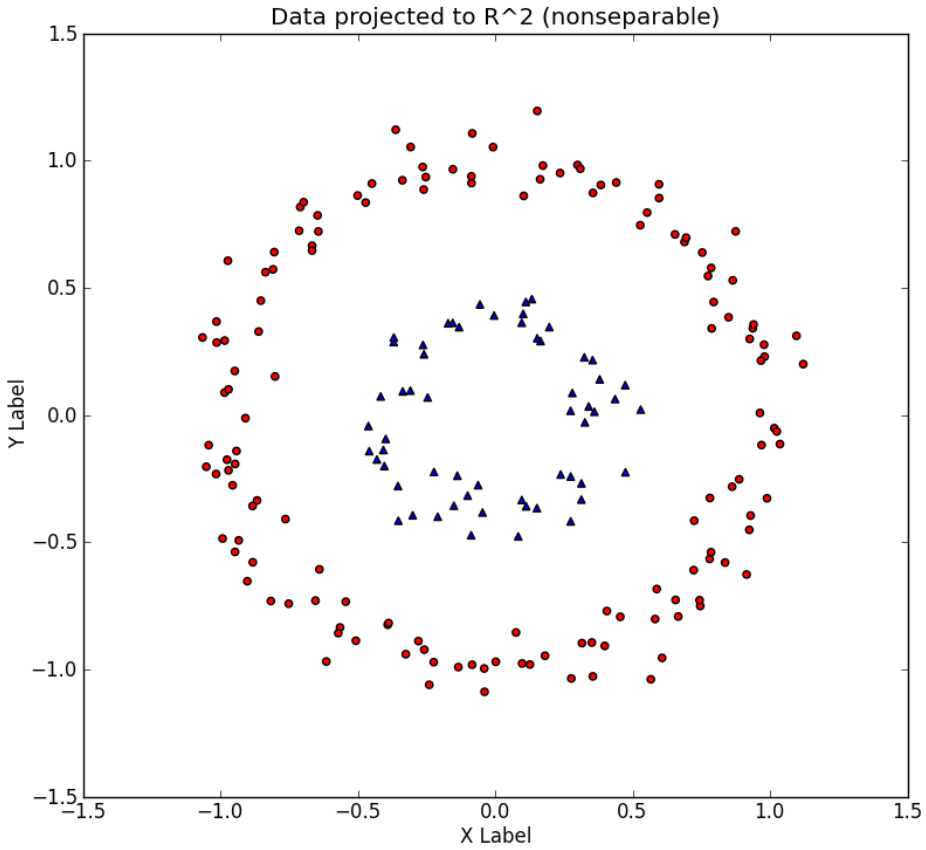
  – задача сводится к SVM с мягким отступом

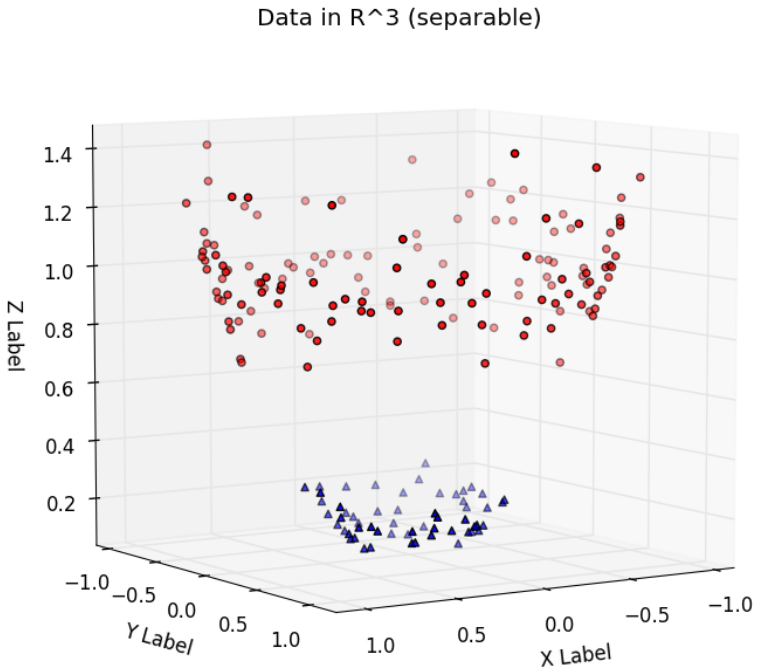
- полиномиальное: $K(\vec{x}_i, \vec{x}_j) = (\gamma \langle \vec{x}_i, \vec{x}_j \rangle + r)^d$

- RBF (Radial Basis Function): $K(\vec{x}_i, \vec{x}_j) = e^{-\gamma \|\vec{x}_i - \vec{x}_j\|^2}$

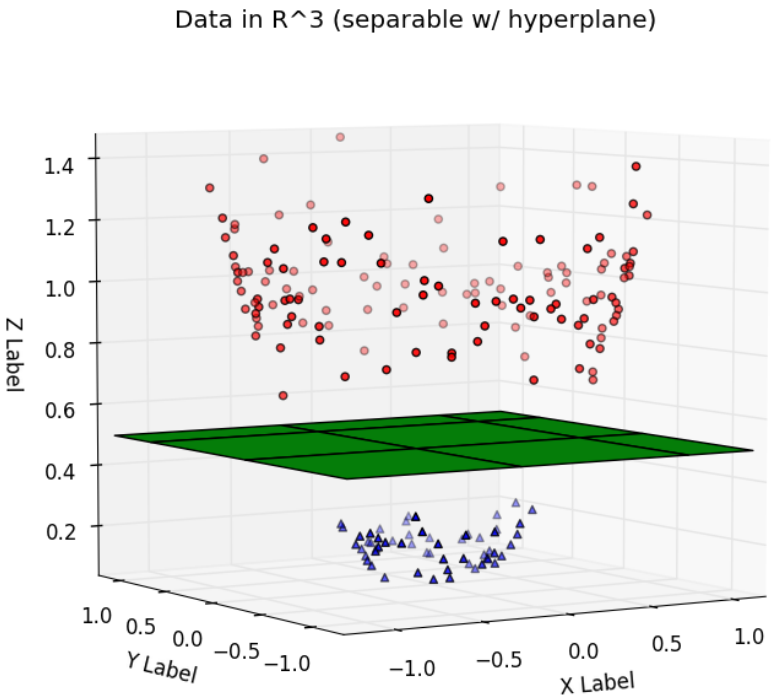
- сигмоидальное: $K(\vec{x}_i, \vec{x}_j) = \tanh(\gamma \langle \vec{x}_i, \vec{x}_j \rangle + r)$

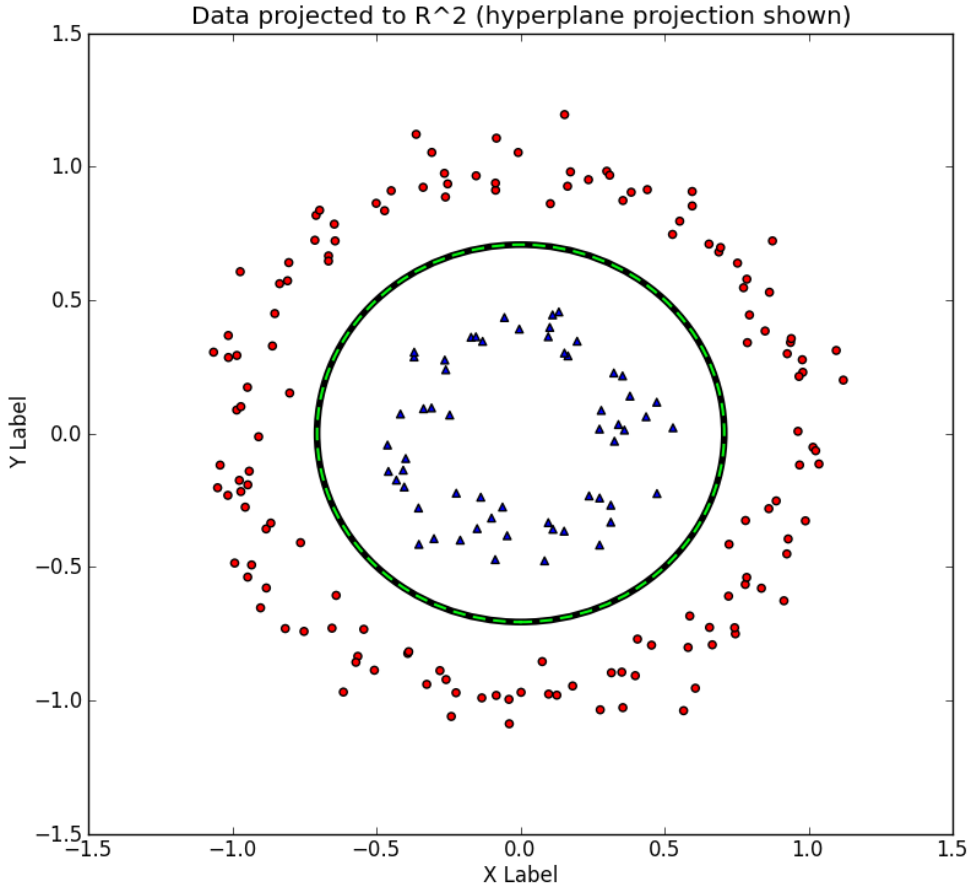
Метод опорных векторов





Метод опорных векторов





Метод опорных векторов

Полиномиальное ядро:

$$K(\vec{x}_i, \vec{x}_j) = \langle \vec{\psi}(\vec{x}_i), \vec{\psi}(\vec{x}_j) \rangle = \left(\langle \vec{x}_i, \vec{x}_j \rangle\right)^2$$

$$\left(\langle \vec{x}_i, \vec{x}_j \rangle\right)^2 = \left(x_{i1}x_{j1} + x_{i2}x_{j2}\right)^2 =$$

$$= \left(x_{i1}x_{j1}\right)^2 + 2x_{i1}x_{j1}x_{i2}x_{j2} + \left(x_{i2}x_{j2}\right)^2 =$$

$$= \left\langle \left(x_{i1}^2, x_{i2}^2, \sqrt{2}x_{i1}x_{i2}\right), \left(x_{j1}^2, x_{j2}^2, \sqrt{2}x_{j1}x_{j2}\right) \right\rangle$$

$$\vec{\psi}(\vec{x}) = \left(x_1^2, x_2^2, \sqrt{2}x_1x_2\right)$$

Многоклассовая классификация

- Multiclass classification: $\mathbb{Y} = \{1, \dots, K\}$
- Варианты решения:
  - сведение к серии бинарных задач
  - построение классификатора сразу для нескольких классов

Многоклассовая классификация

Сведение к серии бинарных задач

1. Один против всех (one-vs-all, one-vs-rest):

- обучим K линейных классификаторов $b_1(x), \dots, b_K(x)$, выдающих оценки принадлежности классам $1, \dots, K$ соответственно
- Например, в случае линейных классификаторов:

$$b_k(\vec{x}) = \langle \vec{w}_k, \vec{x} \rangle + w_{0k}$$

- Классификатор с номером k будем обучать по выборке

$$(x_i, 2[y_i = k] - 1)_{i=1}^{l}$$

- Классификатор учится отличать k-й класс от всех остальных

Многоклассовая классификация

- Итоговый классификатор будет выдавать класс, соответствующий самому уверенному из бинарных алгоритмов:

$$a(x) = \arg\max_{k \in \{1,\dots,K\}} b_k(x) .$$

- Проблема данного подхода заключается в том, что каждый из классификаторов $b_1(x), \dots, b_K(x)$ обучается на своей выборке, и выходы этих классификаторов могут иметь разные масштабы, из-за чего сравнивать их будет неправильно

Многоклассовая классификация

2. Каждый против каждого (one-vs-one):

- Обучим C_K^2 классификаторов $b_{ij}(x)$, $i, j = 1, \ldots, K$, $i \neq j$ (сочетание из K по два):

$$C_K^2 = \frac{K!}{2(K-2)!} = \frac{K(K-1)}{2}$$

- Например, в случае линейных классификаторов:

$$b(\vec{x}) = sign(\langle \vec{w}, \vec{x} \rangle + w_0)$$

Многоклассовая классификация

- Классификатор $b_{ij}(x)$ будем обучать на подвыборке $X_{ij} \subset X$, содержащей только объекты классов i и j:

$$X_{ij} = \{(x_n, y_n) \in X | y_n = i \text{ или } y_n = j\}$$

- Соответственно, классификатор $b_{ij}(x)$ будет выдавать для любого объекта либо класс i, либо класс j
- Чтобы классифицировать новый объект, подадим его на вход каждого из построенных бинарных классификаторов
- Каждый из них проголосует за свой класс; в качестве ответа выберем тот класс, за который наберется больше всего голосов:

$$a(x) = \arg\max_{k \in \{1,\dots,K\}} \sum_{i=1}^{K} \sum_{j \neq i} [b_{ij}(x) = k]$$

Многоклассовая классификация

Многоклассовая логистическая регрессия
(multinomial logistic regression)

- Логистическая регрессия для двух классов:
  - строится линейная модель: $b(x) = (\langle \vec{w}, \vec{x} \rangle + w_0)$
  - вычисляется вероятность: $\sigma(z) = \frac{1}{1+exp(-z)}$

Многоклассовая классификация

- Многоклассовая логистическая регрессия:
  - строятся *K* линейных моделей $b_k(\vec{x}) = \langle \vec{w}_k, \vec{x} \rangle + w_{0k}$
  - каждая модель выдает оценку принадлежности к своему классу и формируется вектор оценок: $\big(b_1(x), \dots, b_K(x)\big)$ – *logits*
  - для получения итоговых результатов оценки нормируются с использованием оператора *SoftMax*:

$$SoftMax(z_1, \dots, z_K) = \left(\frac{exp(z_1)}{\sum_{k=1}^{K} exp(z_k)}, \dots, \frac{exp(z_K)}{\sum_{k=1}^{K} exp(z_k)} \right)$$

  - вероятность k-го класса будет выражаться следующим образом:

$$P(y = k | \vec{x}, \vec{w}) = \frac{exp(\langle \vec{w}_k, \vec{x} \rangle + w_{0k})}{\sum_{j=1}^{K} exp\big(\langle \vec{w}_j, \vec{x} \rangle + w_{0j}\big)}$$

SoftMax

$$SoftMax(z_1, \ldots, z_n) = \left(\frac{e^{z_1}}{\sum_{k=1}^{n} e^{z_k}}, \ldots, \frac{e^{z_n}}{\sum_{k=1}^{n} e^{z_k}} \right)$$

N	Logits	Exp	Probability
1	5	148.4	0.21
2	3	20.1	0.03
3	6.2	492.7	0.68
4	4	54.6	0.08
5	1.5	4.5	0.01
6	-2	0.1	0.00
		720.5	**1.00**