# Линейные модели. Классификация

# Формальная постановка

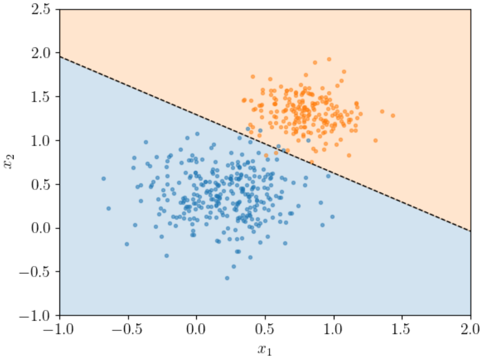
Определить вектор признаков $x$ в один из $K$ классов $C_k$.

# Наивный подход

Классы $0, \ldots, K-1$. Будем решать как задачу регрессии?

# Разделяющая поверхность

Пространство разбивается на классы линейной разделяющей поверхностью. Можем искать эту поверхность — регрессия.




Как размечать и обучать?

# Разделяющие поверхности. Несколько классов

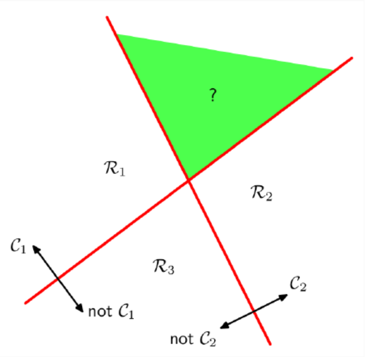



Рис.: Один против всех

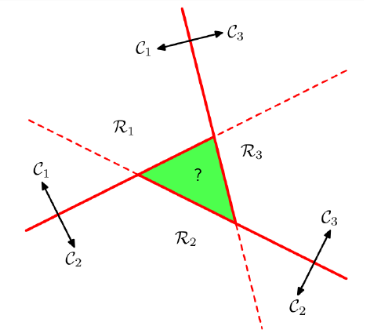
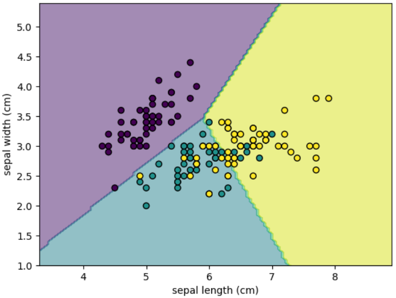



Рис.: Каждый против каждого

# Разделяющие поверхности. Несколько классов




Дискриминантные функции:

$$y_k = w_k^T x, \; k = 0, \ldots, K-1$$

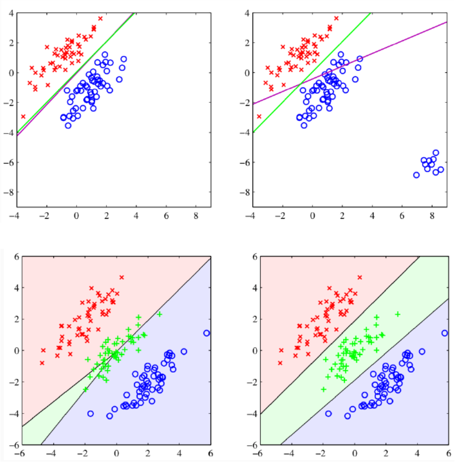
Выбираем класс с максимальным значением $y_k$.

# Обучение

$y_i = (0, \ldots, 0, 1, 0, \ldots, 0)^T$, $y = Wx$.
Решение
$$W = \left(X^T X\right)^{-1} X^T y$$

# Недостатки

Влияние выбросов. Слишком правильные предсказания добавляют штраф



# Дискриминант Фишера

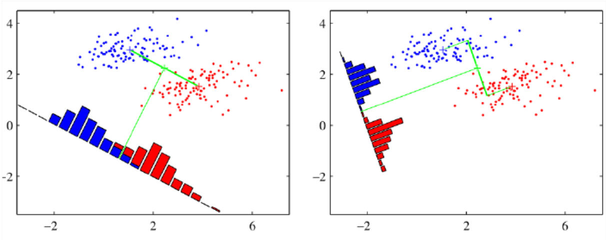
Хотим спроецировать точки $x$ в размерность 1 (на нормаль разделяющей гиперплоскости) так, чтобы в этой размерности 1 они хорошо разделялись. Классификация как метод сокращения размерности.

Пусть даны два класса $C_0$ и $C_1$ с $N_0$ и $N_1$ точками. Найдём серединный перпендикуляр между центрами кластеров $m_0$ и $m_1$:

$$w^T(m_1 - m_0) \to \max$$

$$\|w\| = 1$$

# Дискриминант Фишера




Какая картинка лучше?

# Дискриминант Фишера

Минимизируем перекрытие классов, оптимизируя и проекцию расстояния, и дисперсию.
Дисперсия между классами:

$$S_B = (m_1 - m_0)(m_1 - m_0)^T$$

Дисперсия внутри классов

$$S_W = \sum_{i \in C_0} (x_i - m_0)(x_i - m_0)^T + \sum_{i \in C_1} (x_i - m_1)(x_i - m_1)^T$$

Целевая функция

$$J(w) = \frac{(m_1 - m_0)^2}{s_0^2 + s_1^2} = \frac{w^T S_B w}{w^T S_w w}$$

# Линейный дискриминантный анализ

Представим себе, что точки порождены смесью нормальных распределений (формула Байеса):

$$p(j|x) = \frac{1}{\sqrt{(2\pi^n)^n|\Sigma_j|}} e^{\frac{1}{2}(x-\mu_j)^T\Sigma_j^{-1}(x-\mu_j)} \frac{p(j)}{p(x)},$$

где $p(j)$ — априорная вероятность выбрать класс $j$, $p(x)$ — вероятность точки.
Задача состоит в том, чтобы понять по точке, кто породил этот сигнал.

# Линейный дискриминантный анализ

Пусть все матрицы ковариаций одинаковы $\Sigma_k = \Sigma$. Если зафиксировать $\Sigma$, то границы между классами

$$d_{jk} = \left\{ x | \frac{p(j|x)}{p(k|x)} = 1 \right\}$$

задаются прямыми:

$$d_{jk}(x) = x^T \Sigma^{-1}(\mu_j - \mu_k) - \frac{1}{2}(\mu_j + \mu_k)^T \Sigma^{-1}(\mu_j + \mu_k) + \log \frac{p(j)}{p(k)}.$$

# Линейный дискриминантный анализ

Аналитическое решение:

$$p(j) = \frac{N_j}{N}$$

$$\mu_j = \sum_{i \in C_j} \frac{x_i}{N_j}$$

$$\Sigma = \frac{1}{N - K - 1} \sum_{k=0}^{K-1} \sum_{i \in C_j} (x_i - \mu_k)(x_i - \mu_k)^T$$

# Свойства LDA

- Нормальные распределения в основе
- Решение в аналитическом виде
- Работает даже в далеких от "Гауссовых" ситуаций
- Имеет расширение в квадратичные мономы (QDA)
- Часто рассматривают диагональные $\Sigma_k$ для ускорения вычислений

# Логистическая регрессия

Рассмотрим задачу классификации на два класса $\pm 1$ как задачу регрессии:

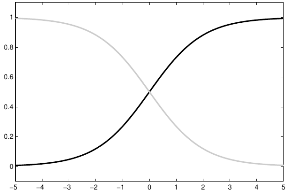
$$y(x) = \operatorname{sign} f(x) = \operatorname{sign} w^T x$$

Хотим, чтобы чем дальше от нуля значение $f(x)$, тем увереннее мы были в классификации объекта $x$.
Тогда для разметки можно взять $+\infty$ для 1 и $-\infty$ для $-1$. Очевидно, что в таком виде оставлять нельзя.

# Логистическая регрессия

Воспользуемся вероятностной постановкой и определим правдоподобие классификации следующим образом (логистическая функция):

$$p(y|x, w) = \frac{1}{1 + e^{-f(x)t}}$$




$$\log \frac{p(1|x)}{p(0|x)} = w^T x$$

# Логистическая регрессия

Максимизируем правдоподобие

$$p(t|X, w) = \prod_{i=1}^{n} p(t_i|x_i, w)$$

Нет аналитического решения, используем численную оптимизацию.

# LDA vs логистическая регрессия

- Есть много точек, для которых нет оценок — LDA
- Есть подозрение на близость к нормальности — LDA
- Хотим использовать prior — LDA
- Во всех остальных случаях логистическая регрессия, особенно если есть много выбросов

# Минимизация эмпирического риска

Рассмотрим задачу классификации на два класса $\pm 1$:

$$y(x) = \operatorname{sign} f(x) = \operatorname{sign} w^T x$$

Отступом алгоритма на объекте $x$ называется величина

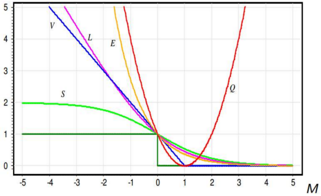
$$M_i = y_i f(x_i)$$

Число ошибок

$$Q(w) = \sum_{i=1}^{n} [M_i(w) < 0]$$

# Минимизация эмпирического риска

$$Q(w) = \sum_{i=1}^{n} [M_i(w) < 0] \le \hat{Q}(w) = \sum_{i=1}^{n} L(M_i(w))$$

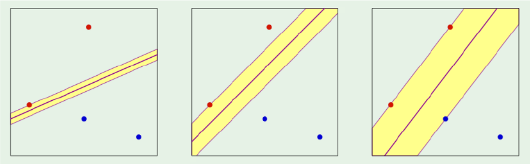



$Q(M) = (1 - M)^2$
$V(M) = (1 - M)_+$
$S(M) = 2(1 + e^M)^{-1}$
$L(M) = \log_2(1 + e^{-M})$
$E(M) = e^{-M}$

Логистическая регрессия: $L(M) = \max(0, 1 - M)$.
Метод опорных векторов: $L(M) = \log(1 + e^{-M})$ с $L_2$-регуляризацией.

# Метод опорных векторов

Пусть выборка линейно-разделима. Максимизируем зазор.



Почему вектора «опорные»?

# Метод опорных векторов

Задача выпуклого квадратичного программирования:

$$\sum_{i=1}^{n} w_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} w_i w_j t_i t_j \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle \to \max_{\boldsymbol{w}}$$

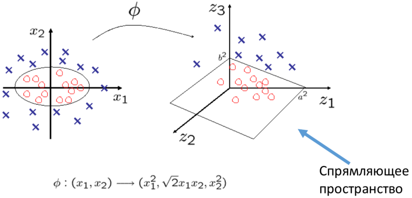
$$\sum_{i=1}^{n} w_i t_i = 0$$

$$0 \leq w_i \leq C$$

Полученную математику используем и в случае неразделимых выборок.

# Метод опорных векторов. Ядра

Если выборка объектов не является линейно разделимой, мы можем предположить, что существует некоторое спрямляющее пространство $H$, вероятно, большей размерности, при переходе в которое выборка станет линейно разделимой.
Не строим $H$ явно, а используем ядра: $< x_i, x_j > \to < \varphi(x_i), \varphi(x_j) >$.




$$\phi : (x_1, x_2) \longrightarrow (x_1^2, \sqrt{2}x_1x_2, x_2^2)$$

$$\left(\frac{x_1}{a}\right)^2 + \left(\frac{x_2}{b}\right)^2 = 1 \longrightarrow \frac{z_1}{a^2} + \frac{z_3}{b^2} = 1$$

Можно делать беспризнаковое распознавание.