# МЕТОД ОПОРНЫХ ВЕКТОРОВ

Сергей Николенко

НИУ ВШЭ — Санкт-Петербург
10 марта 2017 г.

*Random facts*:

- 10 марта 1940 г. родился Чак Норрис, а 10 марта 1957 г. -- Усама бин Ладен
- 10 марта 1876 г. Александр Белл попросил зайти своего помощника Томаса Ватсона зайти к нему в кабинет

# ПРОКЛЯТИЕ РАЗМЕРНОСТИ

# В ПРЕДЫДУЩИХ СЕРИЯХ...

- Теорема Байеса:

$$p(\theta|D) = \frac{p(\theta)p(D|\theta)}{p(D)}.$$

- Две основные задачи байесовского вывода:
  1. найти апостериорное распределение на гипотезах/параметрах:

     $$p(\theta \mid D) \propto p(D|\theta)p(\theta)$$

     (и/или найти максимальную апостериорную гипотезу $\arg\max_\theta p(\theta \mid D)$);
  2. найти апостериорное распределение исходов дальнейших экспериментов:

     $$p(x \mid D) \propto \int_{\theta \in \Theta} p(x \mid \theta)p(D|\theta)p(\theta)\mathrm{d}\theta.$$

# В ПРЕДЫДУЩИХ СЕРИЯХ...

- Мы изучили метод наименьших квадратов для линейной регрессии и метод ближайших соседей...
- ...построили функцию регрессии

$$\hat{f}(\mathbf{x}) = \mathrm{E}_{y|\mathbf{x}'}(y \mid \mathbf{x}' = \mathbf{x})$$

  и оптимальный классификатор

$$\hat{g}(\mathbf{x}) = \arg\min_g \sum_{k=1}^{K} L(g_k, \hat{g}(\mathbf{x}))p(g_k \mid \mathbf{x})...$$

- ...и выяснили, что метод наименьших квадратов – это метод максимального правдоподобия для нормально распределённого шума.

# ПРОКЛЯТИЕ РАЗМЕРНОСТИ

- В размерности 2-3 $k$-NN даёт гораздо более разумные результаты, чем линейная модель, особенно если хорошо выбрать $k$.
- Может быть, нам в этой жизни больше ничего и не нужно?
- Давайте посмотрим, как $k$-NN будет вести себя в более высокой размерности (что очень реалистично).

# ПРОКЛЯТИЕ РАЗМЕРНОСТИ

- Давайте поищем ближайших соседей у точки в единичном гиперкубе. Предположим, что наше исходное распределение равномерное.
- Чтобы покрыть долю $\alpha$ тестовых примеров, нужно (ожидаемо) покрыть долю $\alpha$ объёма, и ожидаемая длина ребра гиперкуба-окрестности в размерности $p$ будет $e_p(\alpha) = \alpha^{1/p}$.
- Например, в размерности $10$ $e_{10}(0.1) = 0.8$, $e_{10}(0.01) = 0.63$, т.е. чтобы покрыть $1\%$ объёма, нужно взять окрестность длиной больше половины носителя по каждой координате!
- Это скажется и на $k$-NN: трудно отвергнуть по малому числу координат, быстрые алгоритмы хуже работают.

# ПРОКЛЯТИЕ РАЗМЕРНОСТИ

- Второе проявление the curse of dimensionality: пусть $N$ точек равномерно распределены в единичном шаре размерности $p$. Тогда среднее расстояние от нуля до точки равно

$$d(p, N) = \left(1 - \frac{1}{2}^{1/N}\right)^{1/p},$$

  т.е., например, в размерности $10$ для $N = 500$ $d \approx 0.52$, т.е. больше половины.

- Большинство точек в результаты ближе к границе носителя, чем к другим точкам, а это для ближайших соседей проблема – придётся не интерполировать внутри существующих точек, а экстраполировать наружу.

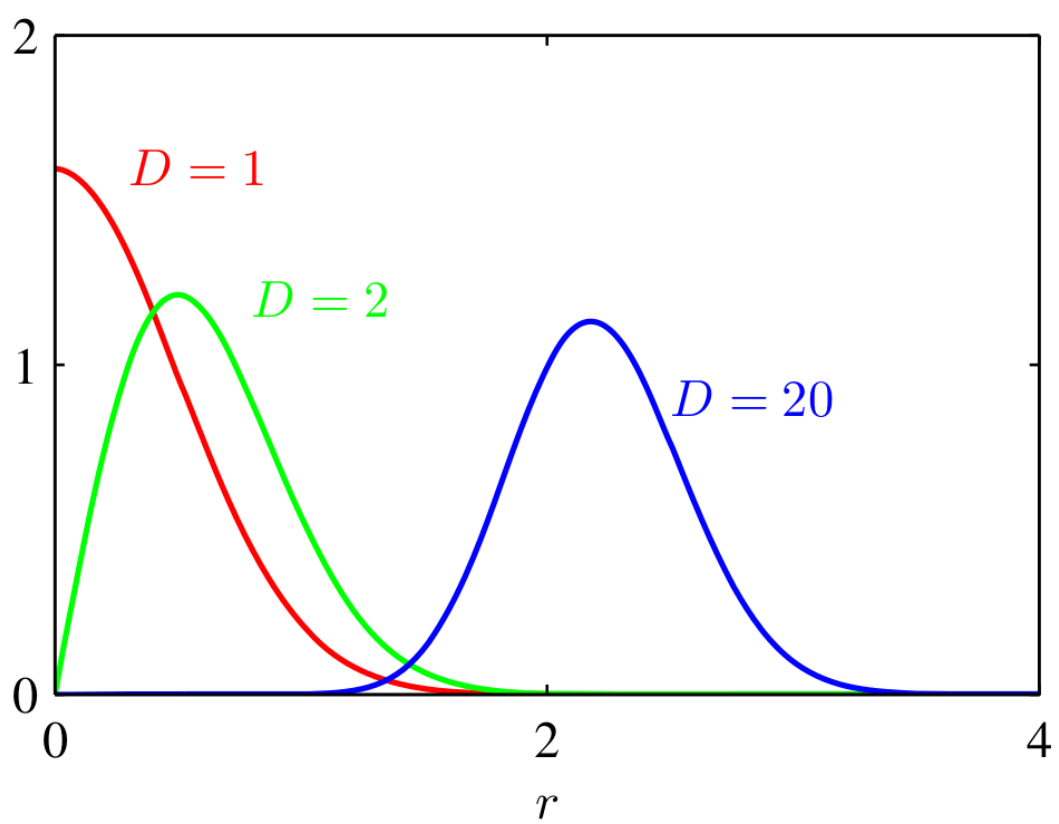
# ПРОКЛЯТИЕ РАЗМЕРНОСТИ

- Третье проявление: проблемы в оптимизации, которые и имел в виду Беллман.
- Если нужно примерно оптимизировать функцию от $d$ переменных, на решётке с шагом $\epsilon$ понадобится примерно $\left(\frac{1}{\epsilon}\right)^d$ вычислений функции.
- В численном интегрировании – чтобы интегрировать функцию с точностью $\epsilon$, нужно тоже примерно $\left(\frac{1}{\epsilon}\right)^d$ вычислений.

# ПРОКЛЯТИЕ РАЗМЕРНОСТИ

- Плотные множества становятся очень разреженными. Например, чтобы получить плотность, создаваемую в размерности $1$ при помощи $N = 100$ точек, в размерности $10$ нужно будет $100^{10}$ точек.
- Поведение функций тоже усложняется с ростом размерности – чтобы строить регрессии в высокой размерности с той же точностью, может потребоваться экспоненциально больше точек, чем в низкой размерности.
- А у линейной модели ничего такого не наблюдается, она не подвержена проклятию размерности.

- Ещё пример: нормально распределённая величина будет сосредоточена в тонкой оболочке.




**Упражнение.** Переведите плотность нормального распределения в полярные координаты и проверьте это утверждение.

# SVM И ЗАДАЧА ЛИНЕЙНОЙ КЛАССИФИКАЦИИ

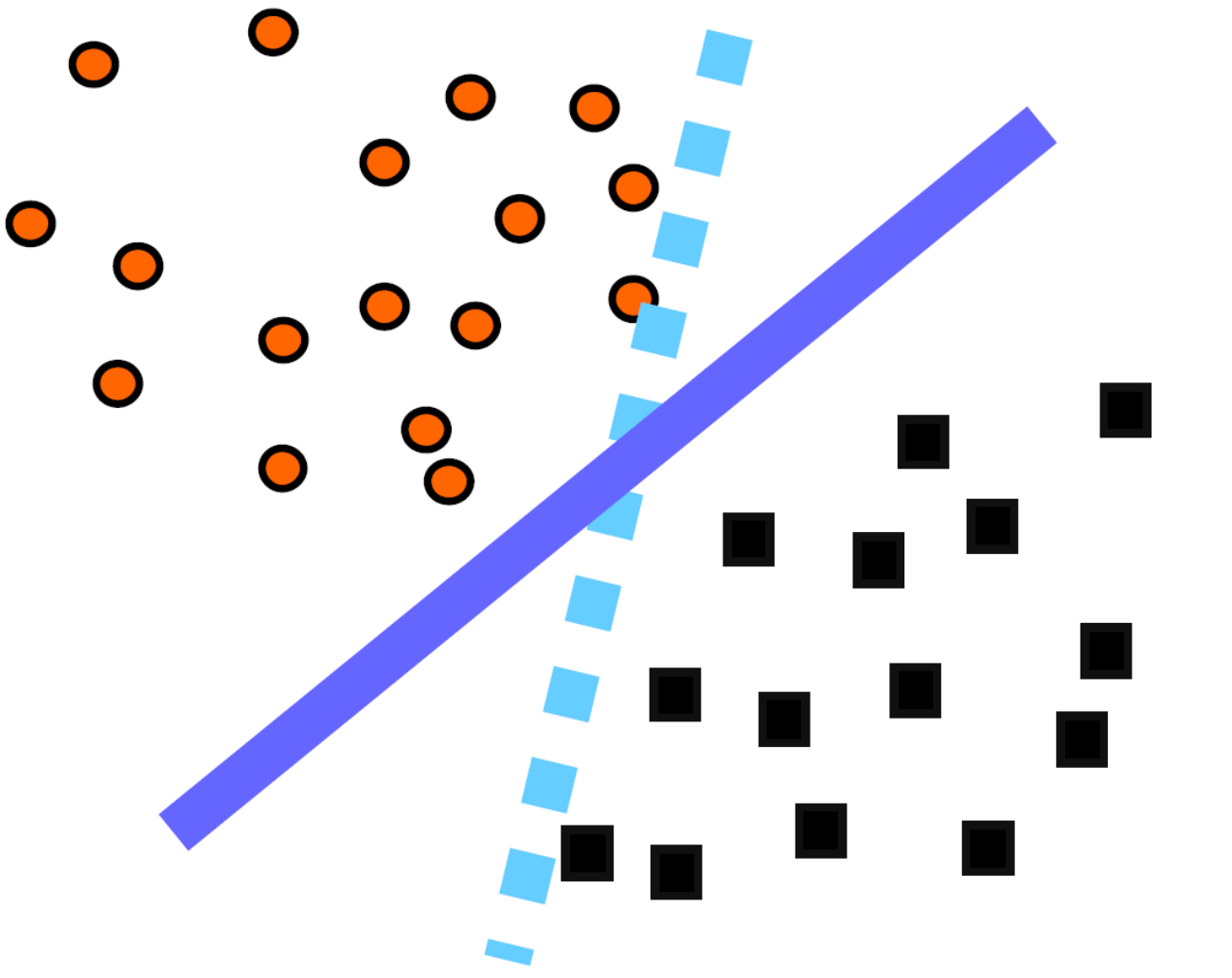
# ПОСТАНОВКА ЗАДАЧИ

- Метод опорных векторов решает задачу классификации.
- Каждый элемент данных — точка в $n$–мерном пространстве $\mathbb{R}^n$.
- Формально: есть точки $x_i$, $i = 1..m$, у точек есть метки $y_i = \pm 1$.
- Мы интересуемся: можно ли разделить данные $(n-1)$–мерной гиперплоскостью, а также хотим найти эту гиперплоскость.
- Это всё?

# ПОСТАНОВКА ЗАДАЧИ

- Нет, ещё хочется научиться разделять этой гиперплоскостью *как можно лучше*.
- То есть желательно, чтобы два разделённых класса лежали как можно дальше от гиперплоскости.
- Практическое соображение: тогда от небольших возмущений в гиперплоскости ничего не испортится.

# ПРИМЕР



# ВЫПУКЛЫЕ ОБОЛОЧКИ

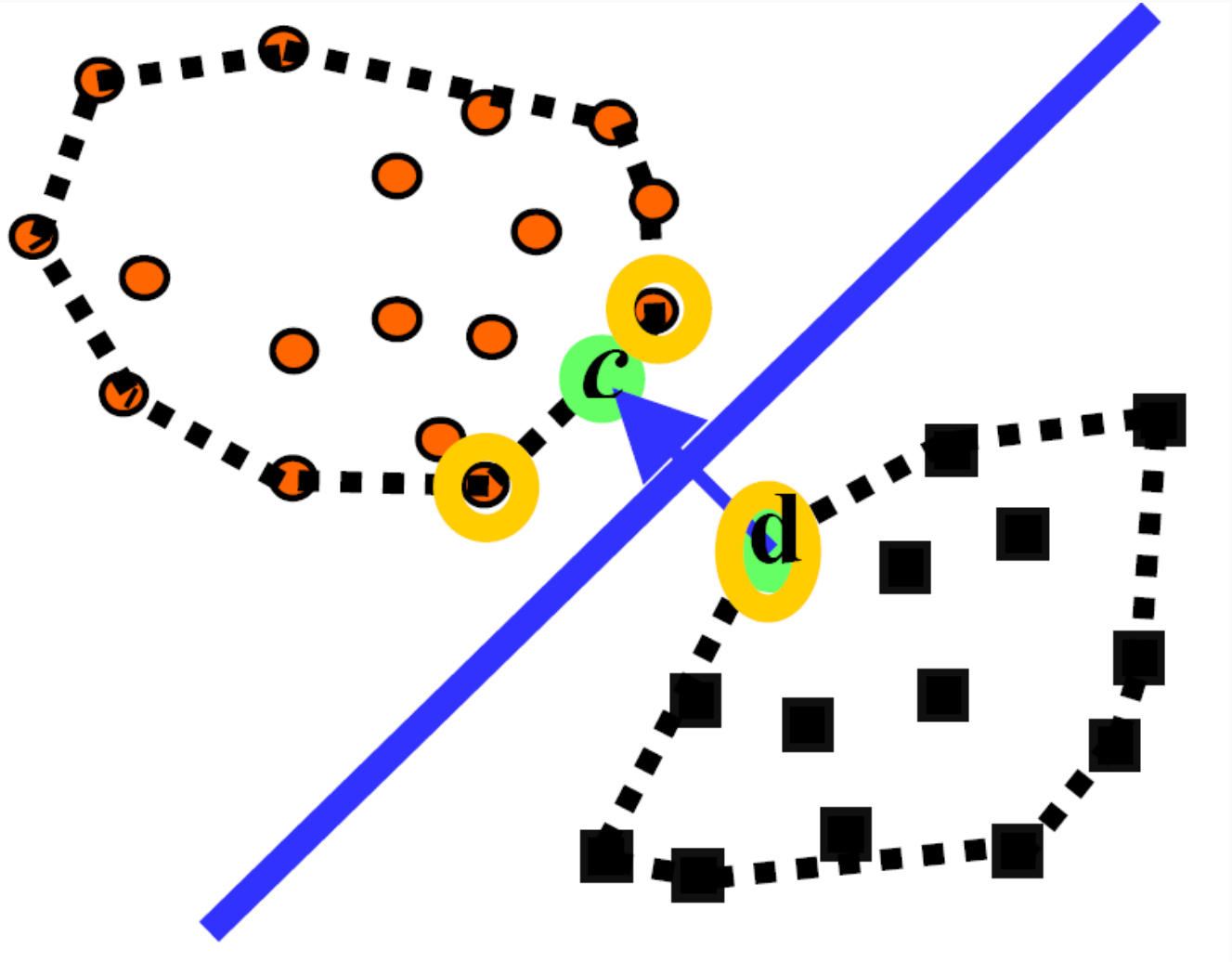
- Один подход: найти две ближайшие точки в выпуклых оболочках данных, а затем провести разделяющую гиперплоскость через середину отрезка.
- Формально это превращается в задачу квадратичной оптимизации:

$$\min_{\alpha} \left\{ ||c-d||^2, \text{ где } c = \sum_{y_i=1} \alpha_i x_i, d = \sum_{y_i=-1} \alpha_i x_i \right\}$$

$$\text{при условии } \sum_{y_i=1} \alpha_i = \sum_{y_i=-1} \alpha_i = 1, \alpha_i \geq 0.$$

- Эту задачу можно решать общими оптимизационными алгоритмами.

# ПРИМЕР



# МАКСИМИЗАЦИЯ ЗАЗОРА

- Другой подход: максимизировать *зазор* (margin) между двумя параллельными опорными плоскостями, затем провести им параллельную на равных расстояниях от них.
- Гиперплоскость называется *опорной* для множества точек $X$, если все точки из $X$ лежат под одну сторону от этой гиперплоскости.
- Формально: расстояние от точки до гиперплоскости $y(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + w_0 = 0$ равно $\frac{|y(\mathbf{x})|}{\|\mathbf{w}\|}$.

# МАКСИМИЗАЦИЯ ЗАЗОРА

- Расстояние от точки до гиперплоскости $y(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + w_0 = 0$ равно $\frac{|y(\mathbf{x})|}{\|\mathbf{w}\|}$.
- Все точки классифицированы правильно: $t_n y(\mathbf{x}_n) > 0$ ($t_n \in \{-1, 1\}$).
- И мы хотим найти

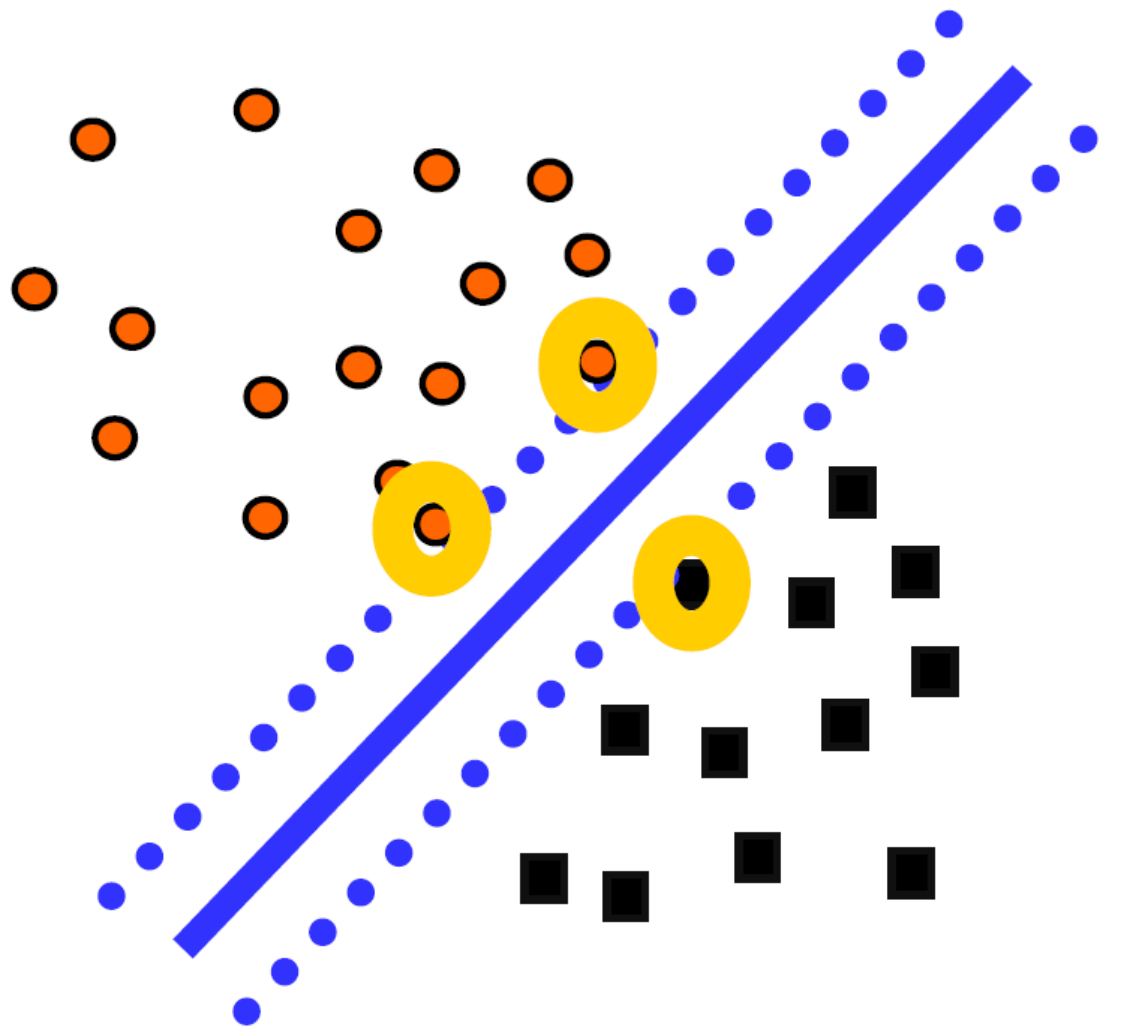
$$\arg\max_{\mathbf{w}, w_0} \min_n \frac{t_n y(\mathbf{x}_n)}{\|\mathbf{w}\|} =$$

$$= \arg\max_{\mathbf{w}, w_0} \left\{ \frac{1}{\|\mathbf{w}\|} \min_n \left[ t_n (\mathbf{w}^\top \mathbf{x}_n + w_0) \right] \right\}.$$

# МАКСИМИЗАЦИЯ ЗАЗОРА

- $\arg\max_{\mathbf{w}, w_0} \left\{ \frac{1}{\|\mathbf{w}\|} \min_n \left[ t_n(\mathbf{w}^\top \mathbf{x}_n + w_0) \right] \right\}$. Сложно.
- Но если перенормировать $\mathbf{w}$, гиперплоскость не изменится.
- Давайте перенормируем так, чтобы $\min_n \left[ t_n(\mathbf{w}^\top \mathbf{x}_n + w_0) \right] = 1$.

# ПРИМЕР



# МАКСИМИЗАЦИЯ ЗАЗОРА

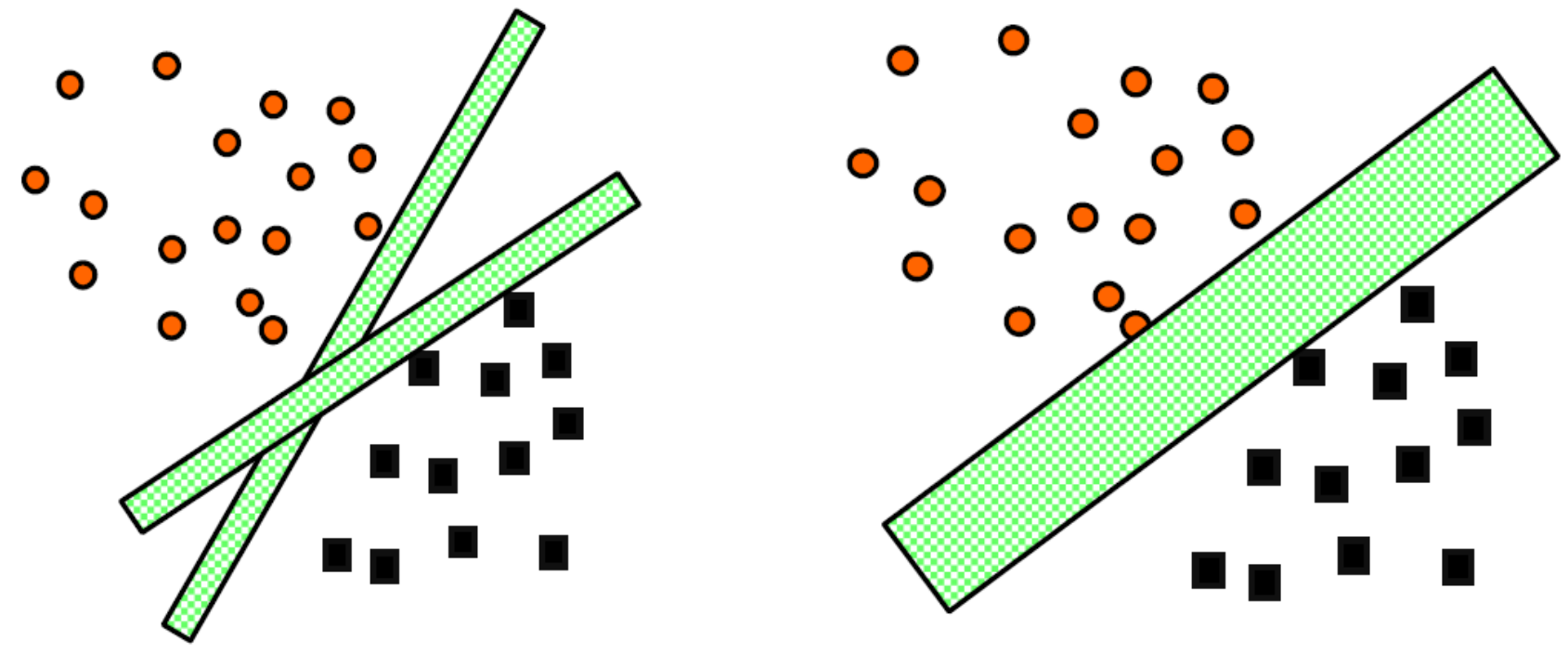
- Получается тоже задача квадратичного программирования:

$$\min_{\vec{w},b} \left\{ \frac{1}{2} ||\mathbf{w}||^2 \right\} \text{ при условии } t_n(\mathbf{w}^\top \mathbf{x}_n + w_0) \geq 1.$$

# РЕЗУЛЬТАТЫ

- Результаты получаются хорошие. Такой подход позволяет находить *устойчивые* решения, что во многом решает проблемы с оверфиттингом и позволяет лучше предсказывать дальнейшую классификацию.
- В каком-то смысле в решениях с «толстыми» гиперплоскостями между данными содержится больше информации, чем в «тонких», потому что «толстых» меньше.
- Это всё можно сформулировать и доказать (позже).

# ПРИМЕР



# ДУАЛЬНЫЕ ЗАДАЧИ

- Напомним, что такое дуальные задачи.
- Прямая задача оптимизации:

$$\min\{f(x)\} \text{ при условии } h(x) = 0,\ g(x) \leq 0,\ x \in X.$$

- Для дуальной задачи вводим параметры $\lambda$, соответствующие равенствам, и $\mu$, соответствующие неравенствам.

# ДУАЛЬНЫЕ ЗАДАЧИ

- Прямая задача оптимизации:

$$\min\{f(x)\} \text{ при условии } h(x) = 0,\ g(x) \leq 0,\ x \in X.$$

- Дуальная задача оптимизации:

$$\min\{\phi(\lambda, \mu)\} \text{ при условии } \mu \geq 0,$$
$$\text{где } \phi(\lambda, \mu) = \inf_{x \in X} \left\{ f(x) + \lambda^\top h(x) + \mu^\top g(x) \right\}.$$

- Тогда, если $(\bar{\lambda}, \bar{\mu})$ – допустимое решение дуальной задачи, а $\bar{x}$ – допустимое решение прямой, то

$$\phi(\bar{\lambda}, \bar{\mu}) = \inf_{x \in X} \left\{ f(x) + \bar{\lambda}^\top h(x) + \bar{\mu}^\top g(x) \right\} \leq$$
$$\leq f(\bar{x}) + \bar{\lambda}^\top h(\bar{x}) + \bar{\mu}^\top g(\bar{x}) \leq f(\bar{x}).$$

- Это называется *слабой дуальностью* (только $\leq$), но во многих случаях достигается и равенство.

# ДУАЛЬНЫЕ ЗАДАЧИ

- Для линейного программирования прямая задача:

$$\min c^\top x \text{ при условии } Ax = b,\ x \in X = \{x \leq 0\}.$$

- Тогда дуальная задача получается так:

$$\phi(\lambda) = \inf_{x \geq 0} \left\{c^\top x + \lambda^\top (b - Ax)\right\} =$$

$$= \lambda^\top b + \inf_{x \geq 0} \left\{(c^\top - \lambda^\top A)x\right\} =$$

$$= \begin{cases} \lambda^\top b, & \text{если } c^\top - \lambda^\top A \geq 0, \\ -\infty & \text{в противном случае.} \end{cases}$$

# ДУАЛЬНЫЕ ЗАДАЧИ

- Для линейного программирования прямая задача:

$$\min\left\{c^\top x\right\} \text{ при условии } Ax = b,\ x \in X = \{x \leq 0\}.$$

- Дуальная задача:

$$\max\left\{b^\top \lambda\right\} \text{ при условии } A^\top \lambda \leq c,\ \lambda \text{ не ограничены.}$$

# ДУАЛЬНЫЕ ЗАДАЧИ

- Для квадратичного программирования прямая задача:

$$\min\left\{\frac{1}{2}x^\top Qx + c^\top x\right\} \text{ при условии } Ax \le b,$$

где $Q$ – положительно полуопределённая матрица (т.е. $x^\top Qx \ge 0$ всегда).

- Дуальная задача (проверьте):

$$\max\left\{\frac{1}{2}\mu^\top D\mu + \mu^\top d - \frac{1}{2}c^\top Q^{-1}c\right\} \text{ при условии } c \ge 0,$$

где $D = -AQ^{-1}A^\top$ (отрицательно определённая матрица), $d = -b - AQ^{-1}c$.

# ДУАЛЬНАЯ ЗАДАЧА К SVM

- В случае SVM надо ввести множители Лагранжа:

$$L(\mathbf{w}, w_0, \alpha) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_n \alpha_n \left[t_n(\mathbf{w}^\top \mathbf{x}_n + w_0) - 1\right], \quad \alpha_n \geq 0.$$

- Берём производные по $\mathbf{w}$ и $w_0$, приравниваем нулю, получаем

$$\mathbf{w} = \sum_n \alpha_n t_n \mathbf{x}_n,$$
$$0 = \sum_n \alpha_n t_n.$$

# ДУАЛЬНАЯ ЗАДАЧА К SVM

- Подставляя в $L(\mathbf{w}, w_0, \alpha)$, получим

$$L(\alpha) = \sum_n \alpha_n - \frac{1}{2} \sum_n \sum_m \alpha_n \alpha_m t_n t_m \left(\mathbf{x}_n^\top \mathbf{x}_m\right)$$

при условии $\alpha_n \geq 0, \sum_n \alpha_n t_n = 0.$

- Это дуальная задача, которая обычно в SVM и используется.

# ПРЕДСКАЗАНИЕ И ККТ

- А для предсказания потом надо посмотреть на знак $y(\mathbf{x})$:

$$y(\mathbf{x}) = \sum_{n=1}^{N} \alpha_n t_n \mathbf{x}^\top \mathbf{x}_n + w_0.$$

- Получилось, что предсказания зависят от всех точек $\mathbf{x}_n$...

# ПРЕДСКАЗАНИЕ И ККТ

- ...но нет. :) Условия ККТ (Karush–Kuhn–Tucker):

$$\begin{aligned} \alpha_n &\geq 0, \\ t_n y(\mathbf{x}_n) - 1 &\geq 0, \\ \alpha_n \left(t_n y(\mathbf{x}_n) - 1\right) &= 0. \end{aligned}$$

- Т.е. реально предсказание зависит от небольшого числа *опорных* векторов, для которых $t_n y(\mathbf{x}_n) = 1$ (они находятся собственно на границе разделяющей поверхности).

# ПОСТАНОВКА ЗАДАЧИ

- Все эти методы работают, когда данные действительно линейно разделимы.
- А что делать, когда их всё-таки немножко не получается разделить?
- Первый вопрос: что делать для первого метода, метода выпуклых оболочек?

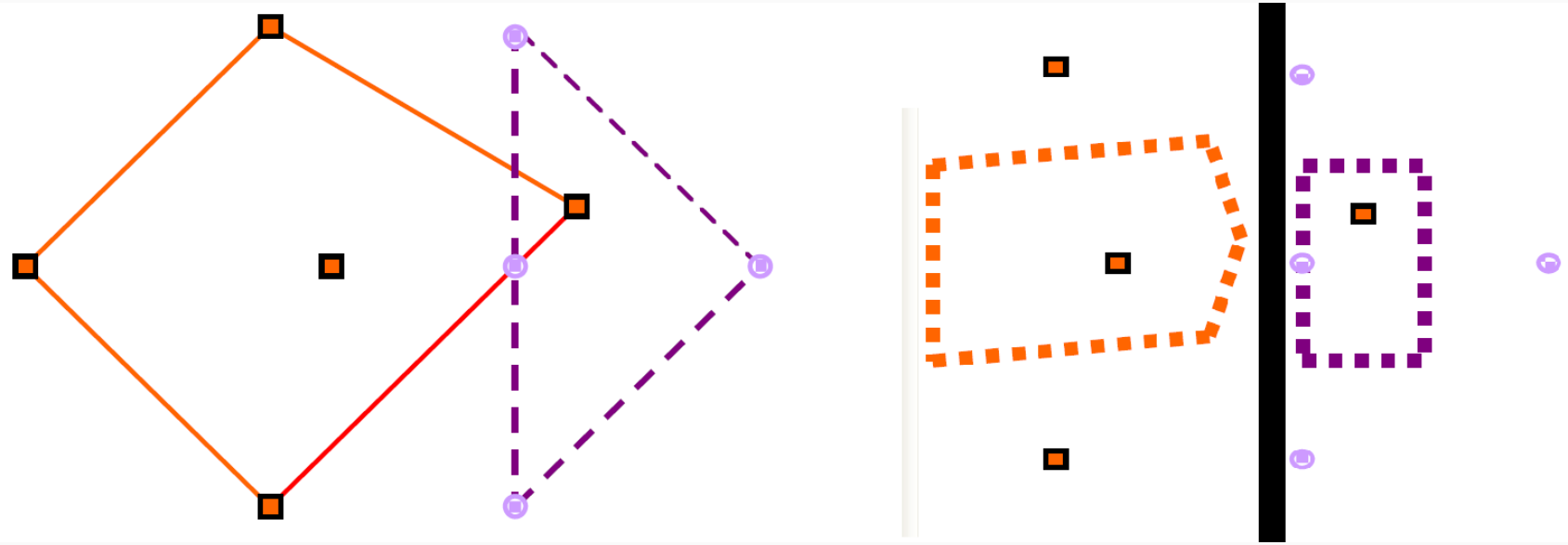
# РЕДУЦИРОВАННЫЕ ВЫПУКЛЫЕ ОБОЛОЧКИ

- Вместо обычных выпуклых оболочек можно рассматривать *редуцированные* (reduced), у которых коэффициенты ограничены не $1$, а сильнее:

$$c = \sum_{y_i=1} \alpha_i x_i, \quad 0 \le \alpha_i \le D.$$

- Тогда для достаточно малых $D$ редуцированные выпуклые оболочки не будут пересекаться.
- И мы будем искать оптимальную гиперплоскость между редуцированными выпуклыми оболочками.

# ПРИМЕР



# ДЛЯ МЕТОДА ОПОРНЫХ ВЕКТОРОВ

- Естественно, для метода опорных векторов тоже надо что-то изменить. Что?

# ДЛЯ МЕТОДА ОПОРНЫХ ВЕКТОРОВ

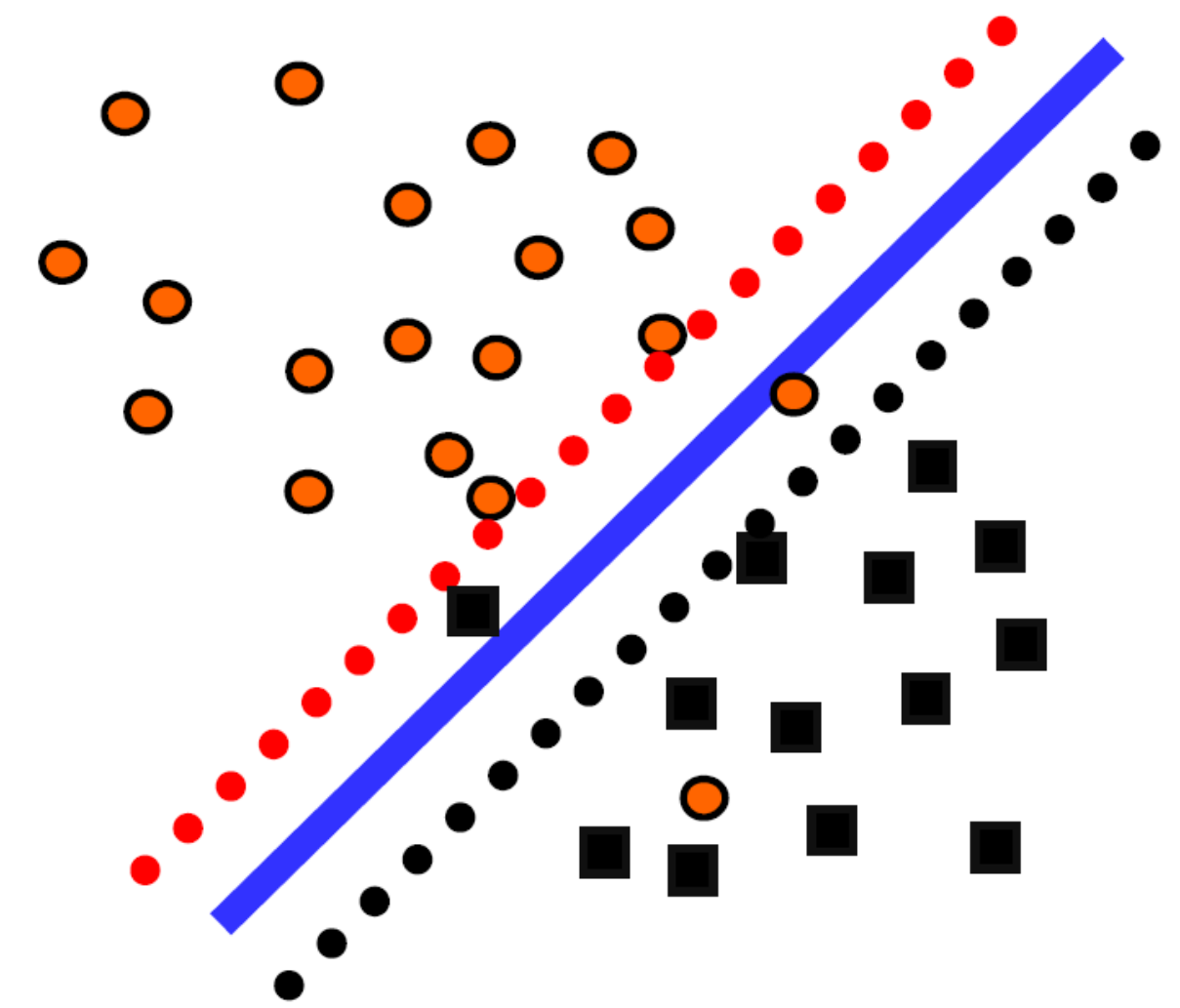
- Естественно, для метода опорных векторов тоже надо что-то изменить. Что?
- Мы просто добавим в оптимизирующуюся функцию неотрицательную ошибку (slack):

$$\min_{\vec{w}, w_0} \left\{ ||\vec{w}||^2 + C \sum_{i=1}^{m} z_i \right\}$$

при условии $t_i(\vec{w} \cdot \vec{x}_i - w_0) + z_i \geq 1.$

- Это прямая задача...

# ПРИМЕР



# ДУАЛЬНАЯ ПЕРЕФОРМУЛИРОВКА

- ...а вот дуальная:

$$\min_{\alpha} \left\{ \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{m} t_i t_j \alpha_i \alpha_j \left( \vec{x}_i \cdot \vec{x}_j \right) - \sum_{i=1}^{m} \alpha_i, \right.$$
$$\left. \text{где } \sum_{i=1}^{m} t_i \alpha_i = 0, \quad 0 \leq \alpha_i \leq C. \right\}$$

- Эта формулировка чаще всего используется в теории SVM.
- Единственное отличие от линейно разделимого случая – верхняя граница $C$ на $\alpha_j$, т.е. на влияние каждой точки.

# ИТОГО

- Метод опорных векторов отлично подходит для линейной классификации.
- Решая задачу квадратичного программирования, мы получаем параметры оптимальной гиперплоскости.
- Точно так же, как и в дуальном случае, если бы мы просто искали середину между выпуклыми оболочками.

# SVM и эмпирический риск

- Ещё один взгляд на SVM — какая вообще задача у любой классификации?
- Мы хотим минимизировать эмпирический риск, то есть число неправильных ответов:

$$\sum_n [y_i \neq t_i] \to \min_{\mathbf{w}}.$$

- И если функция линейная с параметрами $\mathbf{w}$, $w_0$, то это эквивалентно

$$\sum_n \left[t_i \left(\mathbf{x}_n^\top \mathbf{w} - w_0\right) < 0\right] \to \min_{\mathbf{w}}.$$

- Величину $M_i = \mathbf{x}_n^\top \mathbf{w} - w_0$ назовём *отступом* (margin).
- Оптимизировать напрямую сложно…

# SVM и эмпирический риск

- ...поэтому заменим на оценку сверху:

$$\sum_n [M_i < 0] \leq \sum_n (1 - M_i) \to \min_{\mathbf{w}}.$$

- А потом ещё добавим регуляризатор для стабильности:

$$\sum_n [M_i < 0] \leq \sum_n (1 - M_i) + \frac{1}{2C}\|\mathbf{w}\|^2 \to \min_{\mathbf{w}}.$$

- И это снова получилась задача SVM!

# СПАСИБО!

Спасибо за внимание!