Классификаторы

Сергей Николенко

Центр Речевых Технологий, 2012

Центр Речевых Технологий

Outline

Задача классификации

- Теперь классификация: определить вектор $\mathbf{x}$ в один из K классов $\mathcal{C}_k$.
- В итоге у нас так или иначе всё пространство разобьётся на эти классы.
- Т.е. на самом деле мы ищем *разделяющую поверхность* (decision surface, decision boundary).

Задача классификации

- Как кодировать? Бинарная задача – очень естественно, переменная t, $t = 0$ соответствует $\mathcal{C}_1$, $t = 1$ соответствует $\mathcal{C}_2$.
- Оценку t можно интерпретировать как вероятность (по крайней мере, мы постараемся, чтобы было можно).
- Если несколько классов – удобно 1-of-K:

$$\mathbf{t} = (0, \ldots, 0, 1, 0, \ldots)^\top.$$

- Тоже можно интерпретировать как вероятности – или пропорционально им.

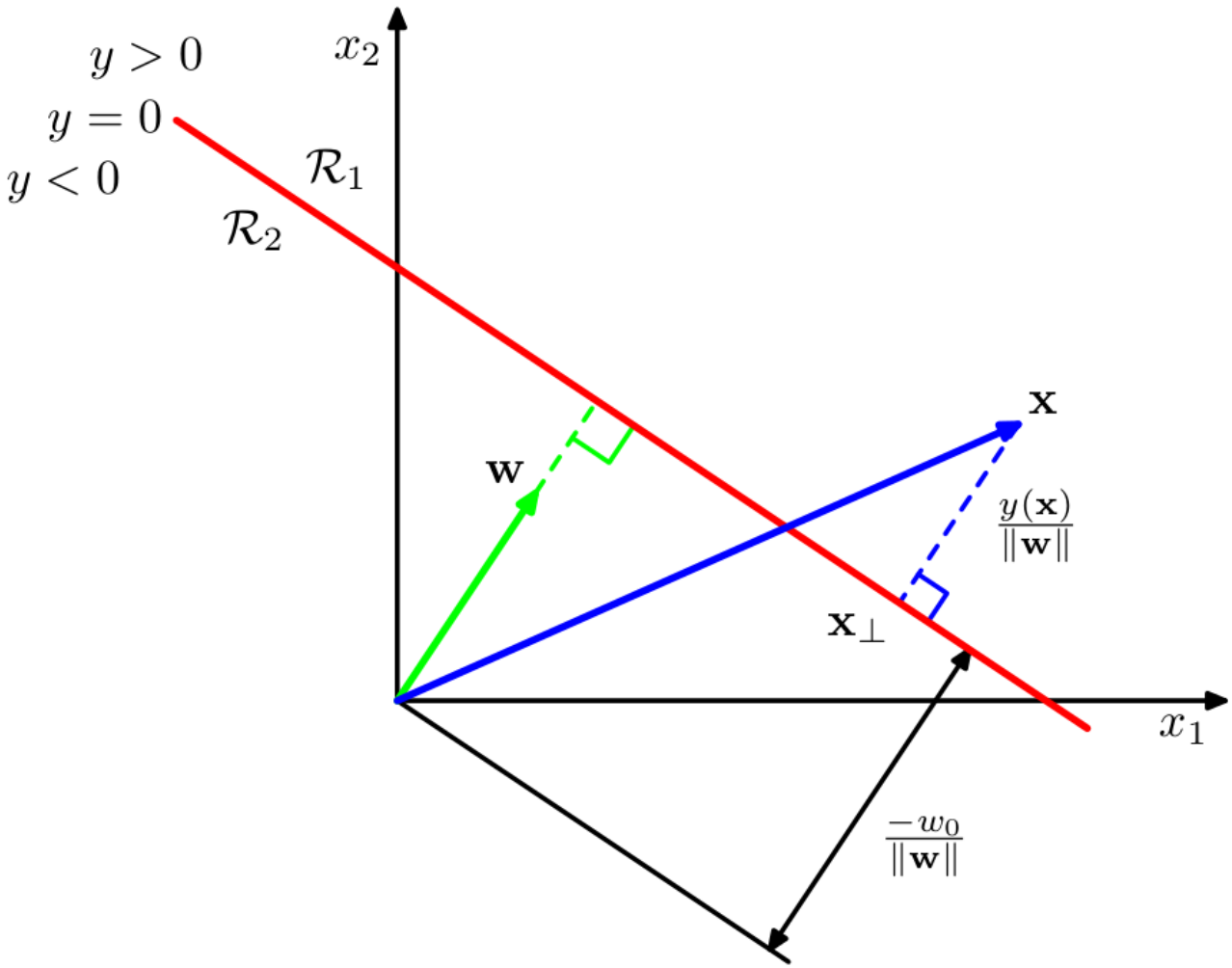
Разделяющая гиперплоскость

- Начнём с геометрии: рассмотрим линейную дискриминантную функцию

$$y(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + w_0.$$

- Это гиперплоскость, и $\mathbf{w}$ – нормаль к ней.
- Расстояние от начала координат до гиперплоскости равно $\frac{-w_0}{\|\mathbf{w}\|}$.
- $y(\mathbf{x})$ связано с расстоянием до гиперплоскости: $d = \frac{y(\mathbf{x})}{\|\mathbf{w}\|}$.

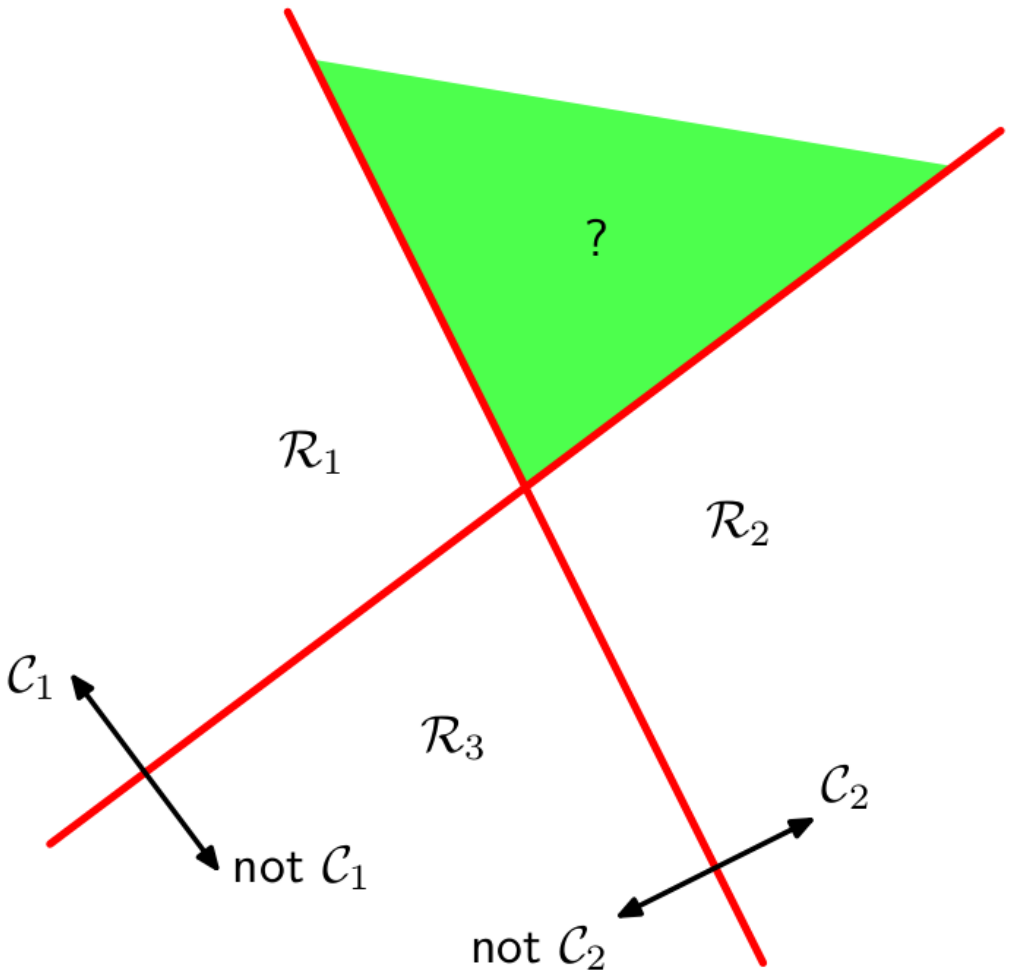
Разделяющая гиперплоскость



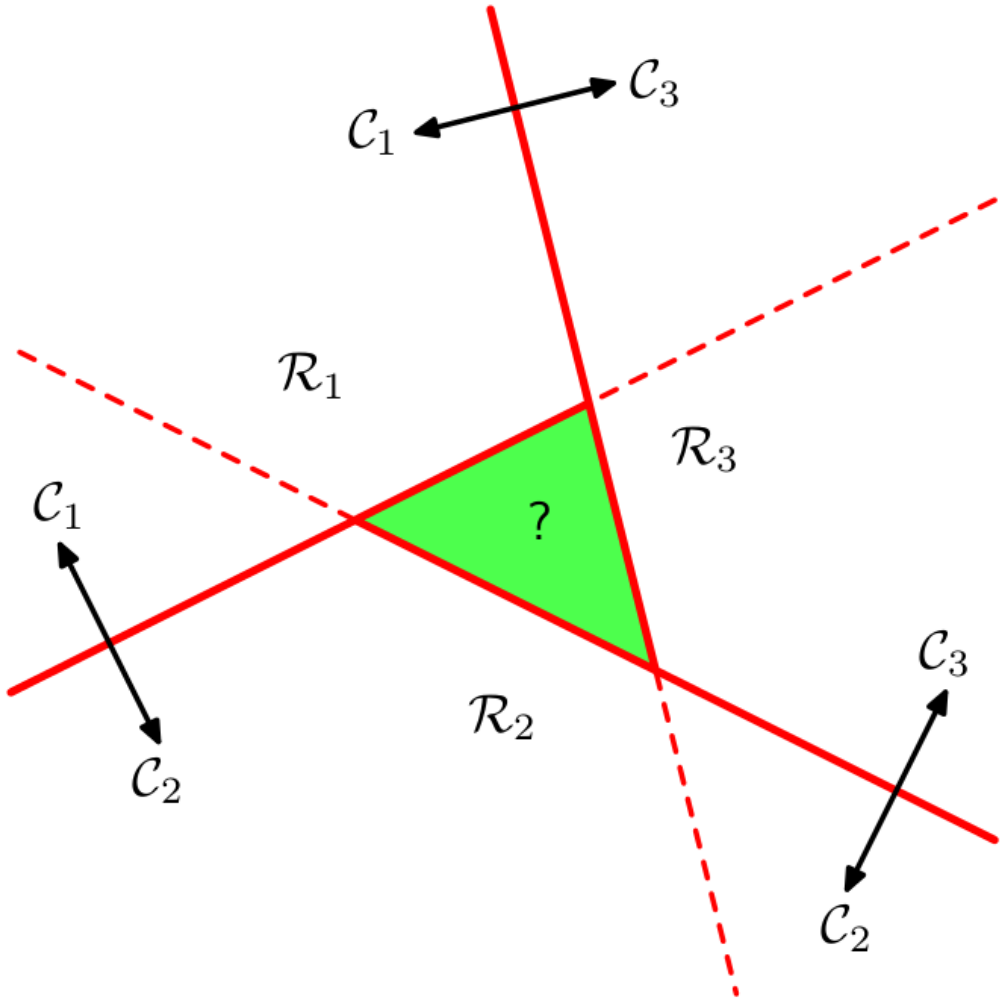
Несколько классов

- С несколькими классами выходит незадача.
- Можно рассмотреть K поверхностей вида «один против всех».
- Можно – $\binom{K}{2}$ поверхностей вида «каждый против каждого».
- Но всё это как-то нехорошо.

Несколько классов



Несколько классов



Несколько классов

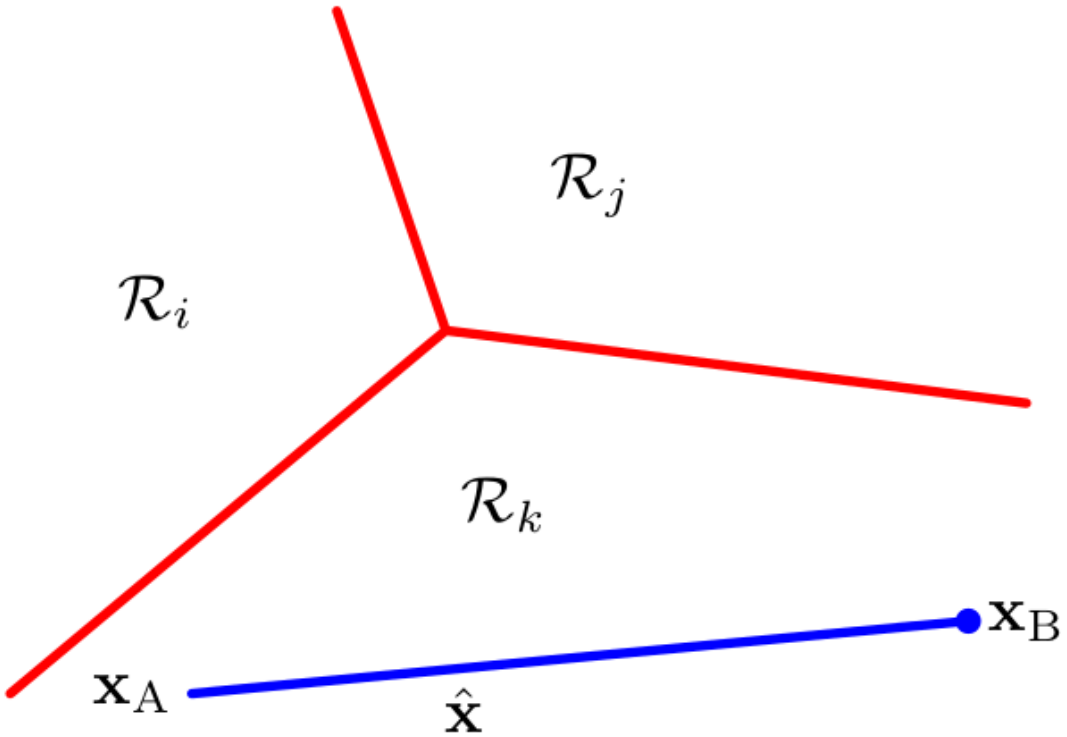
- Лучше рассмотреть единый дискриминант из K линейных функций:

$$y_k(\mathbf{x}) = \mathbf{w}_k^\top \mathbf{x} + w_{k0}.$$

- Классифицировать в $\mathcal{C}_k$, если $y_k(\mathbf{x})$ – максимален.
- Тогда разделяющая поверхность между $\mathcal{C}_k$ и $\mathcal{C}_j$ будет гиперплоскостью вида $y_k(\mathbf{x}) = y_j(\mathbf{x})$:

$$\left(\mathbf{w}_k - \mathbf{w}_j\right)^\top \mathbf{x} + \left(w_{k0} - w_{j0}\right).$$

Несколько классов



Упражнение. Докажите, что области, соответствующие классам, при таком подходе всегда односвязные и выпуклые.

Метод наименьших квадратов

- Мы снова можем воспользоваться методом наименьших квадратов: запишем $y_k(\mathbf{x}) = \mathbf{w}_k^\top \mathbf{x} + w_{k0}$ вместе (спрятав свободный член) как

$$\mathbf{y}(\mathbf{x}) = \mathbf{W}^\top \mathbf{x}.$$

- Можно найти $\mathbf{W}$, оптимизируя сумму квадратов; функция ошибки:

$$E_D(\mathbf{W}) = \frac{1}{2}\mathrm{Tr}\left[(\mathbf{XW} - \mathbf{T})^\top (\mathbf{XW} - \mathbf{T})\right].$$

- Берём производную, решаем...

Метод наименьших квадратов

- ...получается привычное

$$\mathbf{W} = \left(\mathbf{X}^\top \mathbf{X}\right)^{-1} \mathbf{X}^\top \mathbf{T} = \mathbf{X}^\dagger \mathbf{T},$$

  где $\mathbf{X}^\dagger$ – псевдообратная Мура-Пенроуза.
- Теперь можно найти и дискриминантную функцию:

$$\mathbf{y}(\mathbf{x}) = \mathbf{W}^\top \mathbf{x} = \mathbf{T}^\top \left(\mathbf{X}^\dagger\right)^\top \mathbf{x}.$$

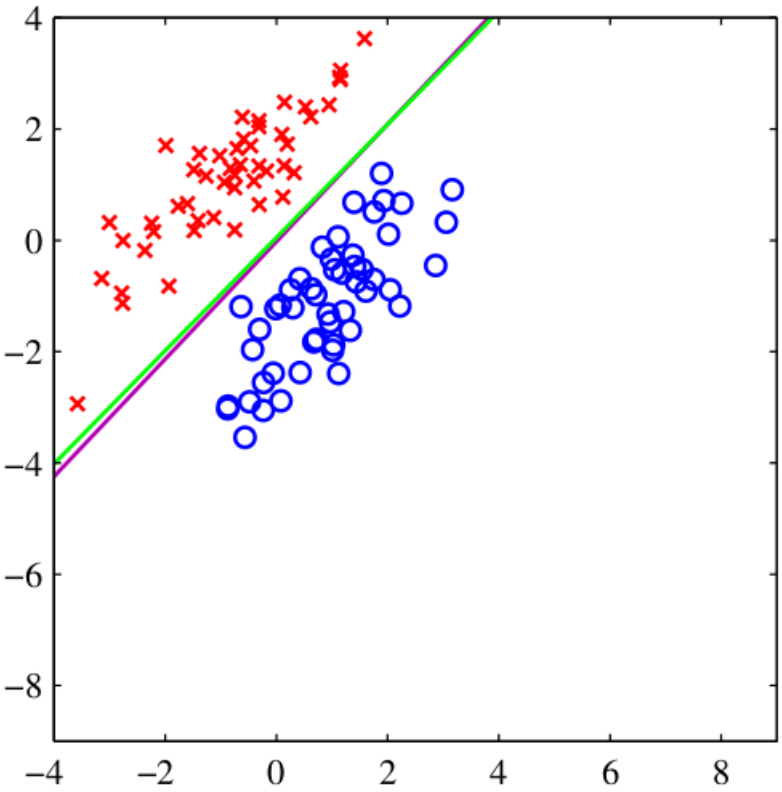
Метод наименьших квадратов

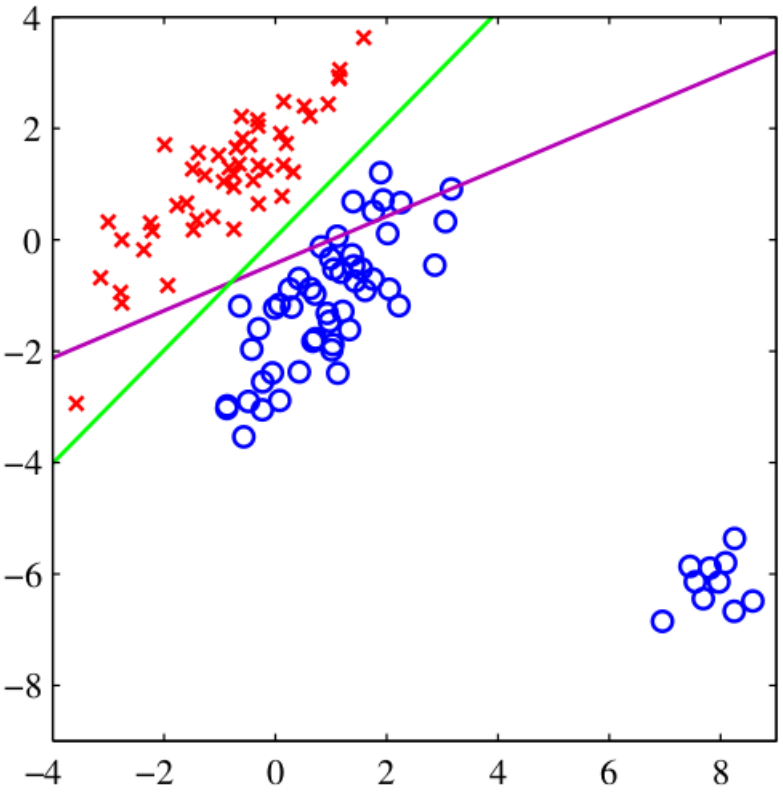
- Это решение сохраняет линейность.

Упражнение. Докажите, что в схеме кодирования 1-of-K предсказания $y_k(\mathbf{x})$ для разных классов при любом $\mathbf{x}$ будут давать в сумме 1. Почему они всё-таки не будут разумными оценками вероятностей?

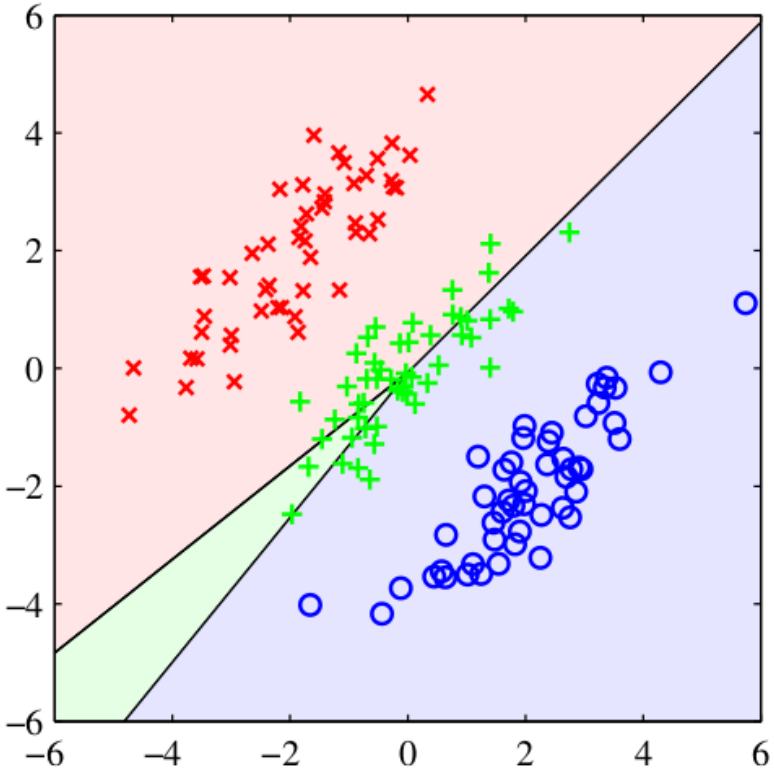
- Проблемы наименьших квадратов:
  - outliers плохо обрабатываются;
  - «слишком правильные» предсказания добавляют штраф.

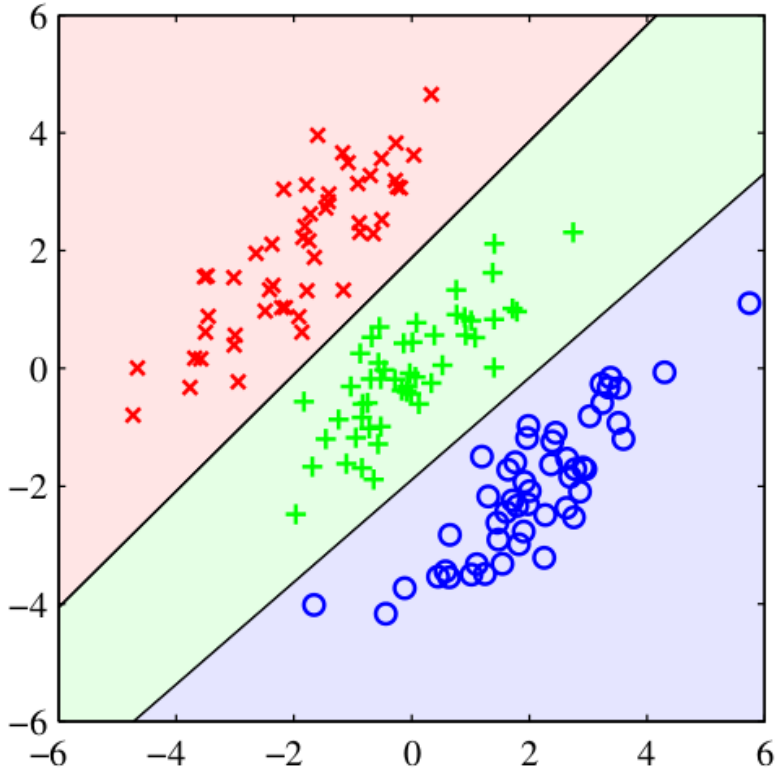
Проблемы наименьших квадратов





Проблемы наименьших квадратов





Проблемы наименьших квадратов

- Почему так? Почему наименьшие квадраты так плохо работают?

Проблемы наименьших квадратов

- Почему так? Почему наименьшие квадраты так плохо работают?
- Они предполагают гауссовское распределение ошибки.
- Но, конечно, распределение у бинарных векторов далеко не гауссово.

Линейный дискриминант Фишера

- Другой взгляд на классификацию: в линейном случае мы хотим спроецировать точки в размерность 1 (на нормаль разделяющей гиперплоскости) так, чтобы в этой размерности 1 они хорошо разделялись.
- Т.е. классификация – это такой метод радикального сокращения размерности.
- Давайте посмотрим на классификацию с этих позиций и попробуем добиться оптимальности в каком-то смысле.

Линейный дискриминант Фишера

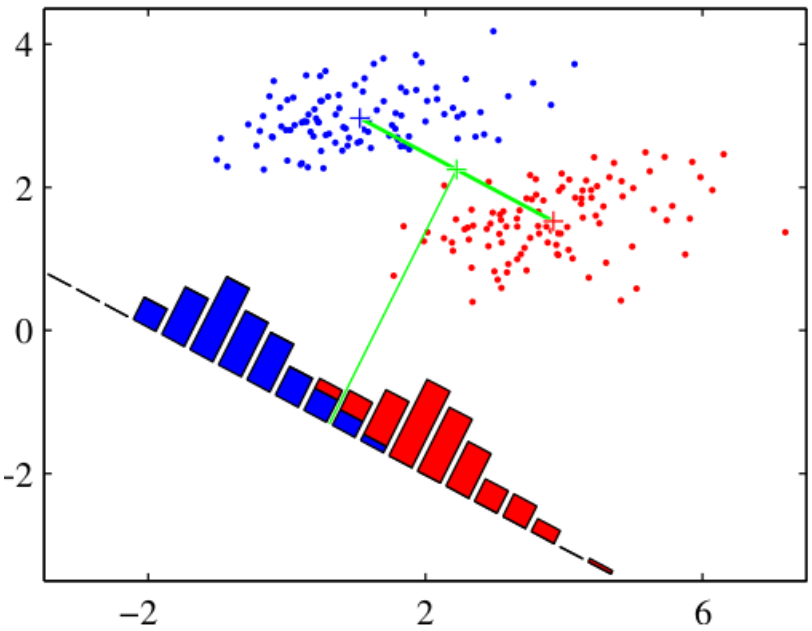
- Рассмотрим два класса $\mathcal{C}_1$ и $\mathcal{C}_2$ с N_1 и N_2 точками.
- Первая идея – надо найти серединный перпендикуляр между центрами кластеров

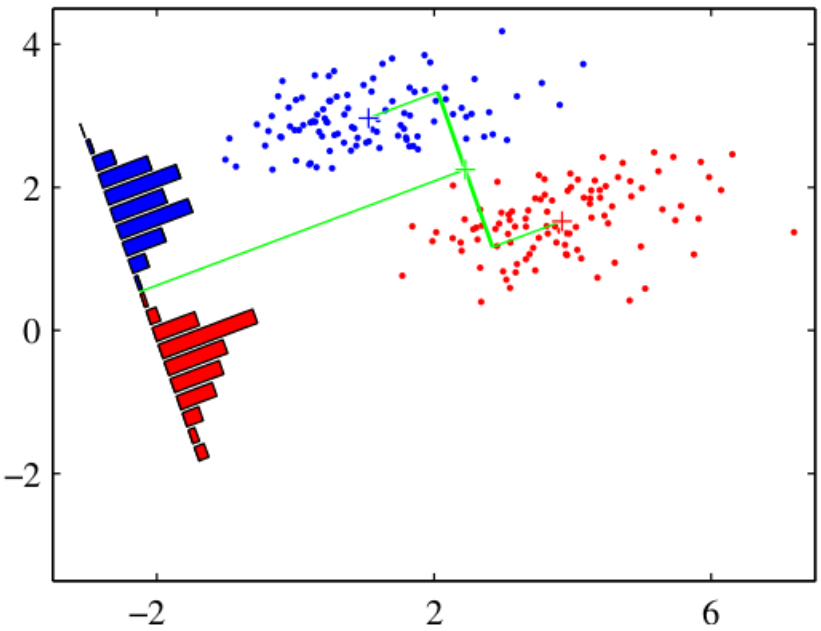
$$\mathbf{m}_1 = \frac{1}{N_1} \sum_{\mathcal{C}_1} \mathbf{x}, \text{ и } \mathbf{m}_2 = \frac{1}{N_2} \sum_{\mathcal{C}_2} \mathbf{x},$$

т.е. максимизировать $\mathbf{w}^\top (\mathbf{m}_2 - \mathbf{m}_1)$.

- Надо ещё добавить ограничение $\|\mathbf{w}\| = 1$, но всё равно не ахти как работает.

Линейный дискриминант Фишера





Чем левая картинка хуже правой?

Линейный дискриминант Фишера

- Слева больше дисперсия каждого кластера.
- Идея: минимизировать перекрытие классов, оптимизируя и проекцию расстояния, и дисперсию.
- Выборочные дисперсии в проекции: для $y_n = \mathbf{w}^\top \mathbf{x}_n$

$$s_1 = \sum_{n \in \mathcal{C}_1} (y_n - m_1)^2 \text{ и } s_1 = \sum_{n \in \mathcal{C}_2} (y_n - m_2)^2 .$$

Линейный дискриминант Фишера

- Критерий Фишера:

$$J(\mathbf{w}) = \frac{(m_2 - m_1)^2}{s_1^2 + s_2^2} = \frac{\mathbf{w}^\top \mathbf{S}_B \mathbf{w}}{\mathbf{w}^\top \mathbf{S}_W \mathbf{w}}, \text{ где}$$

$$\mathbf{S}_B = (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^\top,$$

$$\mathbf{S}_W = \sum_{n \in \mathcal{C}_1} (\mathbf{x}_n - \mathbf{m}_1)(\mathbf{x}_n - \mathbf{m}_1)^\top + \sum_{n \in \mathcal{C}_2} (\mathbf{x}_n - \mathbf{m}_2)(\mathbf{x}_n - \mathbf{m}_2)^\top$$

(between-class covariance и within-class covariance).

- Дифференцируя по $\mathbf{w}$...

Линейный дискриминант Фишера

- ...получим, что $J(\mathbf{w})$ максимален при

$$\left(\mathbf{w}^\top \mathbf{S}_B \mathbf{w}\right) \mathbf{S}_W \mathbf{w} = \left(\mathbf{w}^\top \mathbf{S}_W \mathbf{w}\right) \mathbf{S}_B \mathbf{w}.$$

- Т.к. $\mathbf{S}_B = (\mathbf{m}_2 - \mathbf{m}_1)(\mathbf{m}_2 - \mathbf{m}_1)^\top$, $\mathbf{S}_B \mathbf{w}$ всё равно будет в направлении $\mathbf{m}_2 - \mathbf{m}_1$, а длина $\mathbf{w}$ нас не интересует.
- Поэтому получается

$$\mathbf{w} \propto \mathbf{S}_W^{-1} (\mathbf{m}_2 - \mathbf{m}_1).$$

- В итоге мы выбрали направление проекции, и осталось только разделить данные на этой проекции.

Линейный дискриминант Фишера

- Любопытно, что дискриминант Фишера тоже можно получить из наименьших квадратов.
- Давайте для класса $\mathcal{C}_1$ выберем целевое значение $\frac{N_1+N_2}{N_1}$, а для класса $\mathcal{C}_2$ возьмём $-\frac{N_1+N_2}{N_2}$.

Упражнение. Докажите, что при таких целевых значениях наименьшие квадраты – это дискриминант Фишера.

Линейный дискриминант Фишера

- А что будет с несколькими классами? Рассмотрим $\mathbf{y} = \mathbf{W}^\top \mathbf{x}$, обобщим внутреннюю дисперсию как

$$\mathbf{S}_W = \sum_{k=1}^{K} \mathbf{S}_k = \sum_{k=1}^{K} \sum_{n \in \mathcal{C}_k} (\mathbf{x}_n - \mathbf{m}_k)(\mathbf{x}_n - \mathbf{m}_k)^\top .$$

- Чтобы обобщить внешнюю (межклассовую) дисперсию, просто возьмём остаток полной дисперсии

$$\mathbf{S}_T = \sum_{n} (\mathbf{x}_n - \mathbf{m})(\mathbf{x}_n - \mathbf{m})^\top ,$$

$$\mathbf{S}_B = \mathbf{S}_T - \mathbf{S}_W .$$

Линейный дискриминант Фишера

- Обобщить критерий можно разными способами, например:

$$J(\mathbf{W}) = \mathrm{Tr}\left[\mathbf{s}_W^{-1}\mathbf{s}_B\right],$$

где $\mathbf{s}$ – ковариации в пространстве проекций на $\mathbf{y}$:

$$\mathbf{s}_W = \sum_{k=1}^{K} \sum_{n \in \mathcal{C}_k} (\mathbf{y}_n - \boldsymbol{\mu}_k)(\mathbf{y}_n - \boldsymbol{\mu}_k)^\top,$$

$$\mathbf{s}_B = \sum_{k=1}^{K} N_k (\boldsymbol{\mu}_k - \boldsymbol{\mu})(\boldsymbol{\mu}_k - \boldsymbol{\mu})^\top,$$

где $\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{n \in \mathcal{C}_k} \mathbf{y}_n$.

Центр Речевых Технологий

Outline

Перцептрон

- Другой пример линейного дискриминанта – *перцептрон* (Rosenblatt, 1962).
- Классифицируем вектор $\mathbf{x}$, сначала выделив признаки $\boldsymbol{\phi}(\mathbf{x})$, а затем классифицируя как

$$y(\mathbf{x}) = \text{sign}(\mathbf{w}^\top \boldsymbol{\phi}(\mathbf{x})),$$

где $\text{sign}(x) = +1$ для $x \geq 0$ и -1 для $x < 0$.

- Целевые значения – $t = 1$ для $\mathcal{C}_1$ и $t = -1$ для $\mathcal{C}_2$.

Перцептрон

- Было бы хорошо выбрать функцию ошибки как число неверно классифицированных примеров.
- Но тогда получится кусочно-гладкая функция ошибки с кучей разрывов – непонятно, как обучать $\mathbf{w}$.
- Поэтому мы используем критерий перцептрона:

$$E_P(\mathbf{w}) = -\sum_{n \in \mathcal{M}} \mathbf{w}^\top \boldsymbol{\phi}(\mathbf{x}_n) t_n,$$

где $\mathcal{M}$ – неверно классифицированные примеры (т.е. мы минимизируем суммарное отклонение в неправильную сторону).

- Получилась кусочно-линейная функция.

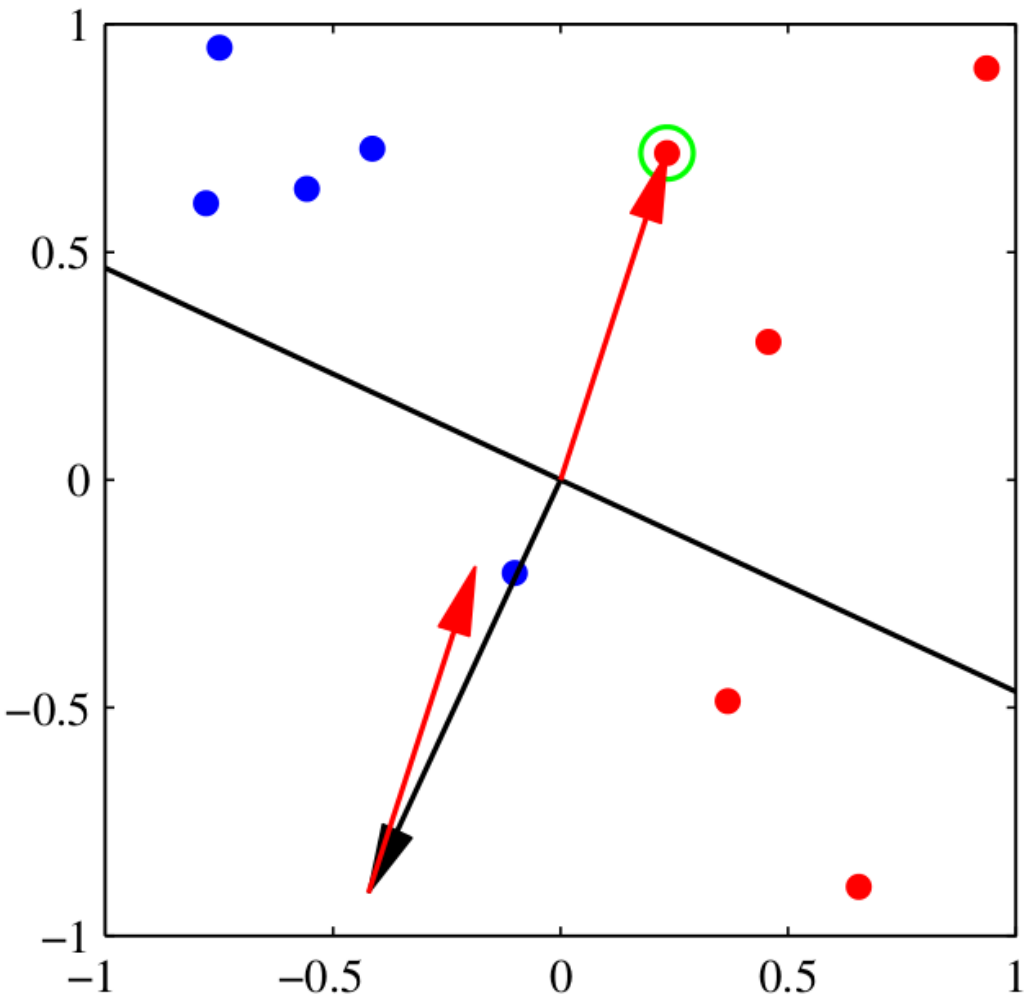
Перцептрон

- Оптимизировать – градиентным спуском:

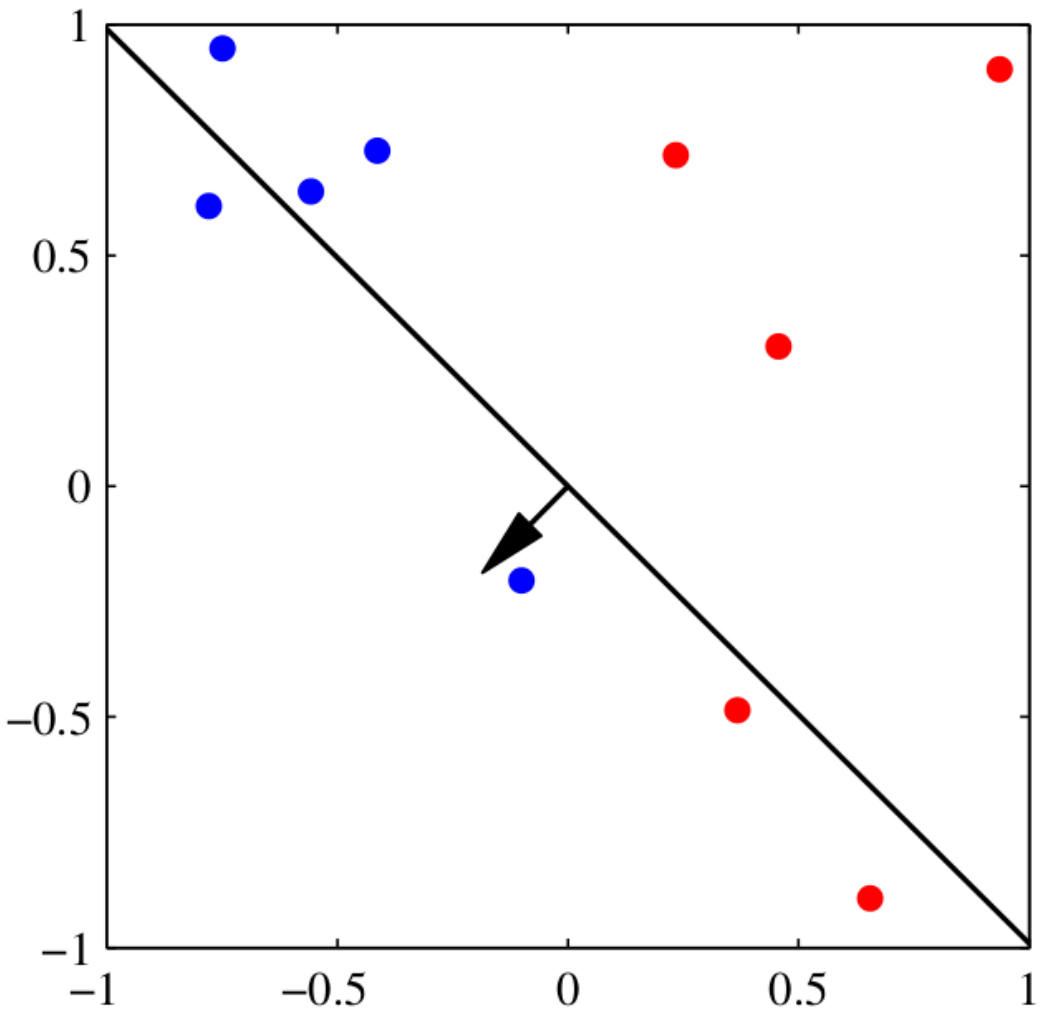
$$\mathbf{w}^{(\tau+1)} = \mathbf{w}^{(\tau)} - \eta \nabla E_P(\mathbf{w}) = \mathbf{w}^{(\tau)} + \eta \boldsymbol{\phi}(\mathbf{x}_n) t_n.$$

- Т.е. мы идём по примерам, и для каждого $\mathbf{x}_n$:
  - если он классифицирован правильно, ничего не меняем;
  - если неправильно, добавляем или вычитаем $\eta \boldsymbol{\phi}(\mathbf{x}_n)$ к $\mathbf{w}$.
- Ошибка от $\mathbf{x}_n$ при этом, очевидно, уменьшается, но, конечно, совершенно никто не гарантирует, что при этом не увеличится ошибка от других примеров.

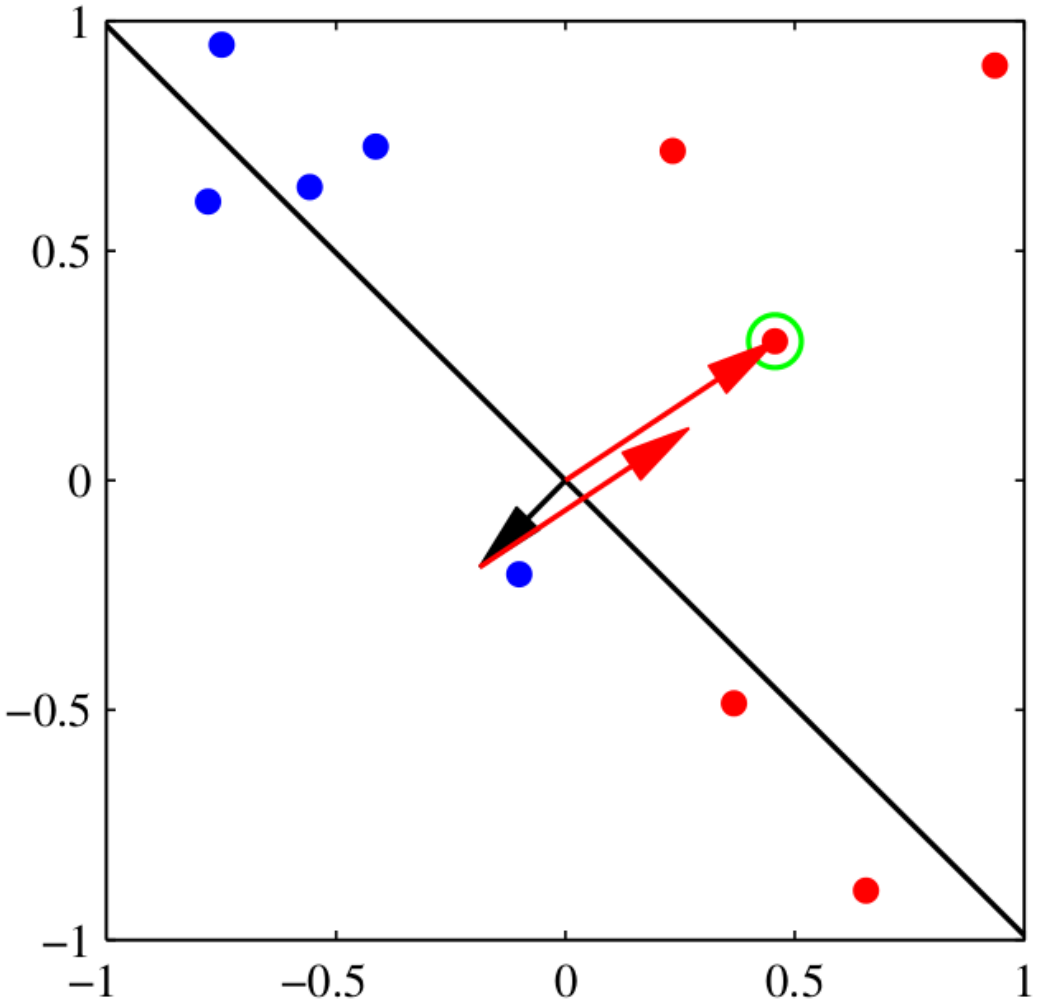
Обучение перцептрона



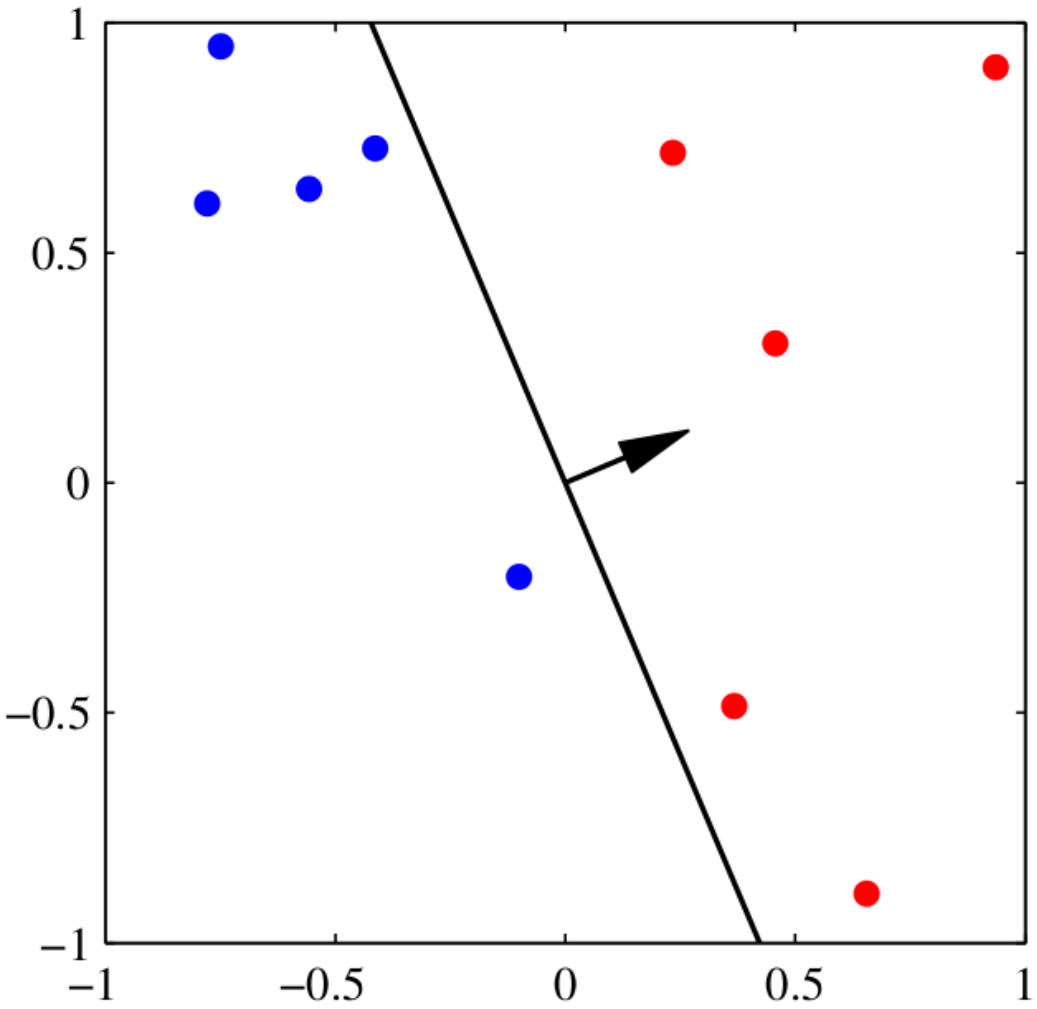
Обучение перцептрона



Обучение перцептрона



Обучение перцептрона



Сходимость

Этот алгоритм сходится всегда, когда это возможно.

Теорема

Если конечное множество точек $\mathcal{C}_1 \subset \{0,1\}^n$ можно в $\{0,1\}^n$ отделить гиперплоскостью от конечного множества точек $\mathcal{C}_2 \subset \{0,1\}^n$, то алгоритм обучения перцептрона за конечное количество шагов выдаёт параметры перцептрона, который успешно разделяет множества $\mathcal{C}_1$ и $\mathcal{C}_2$.

Доказательство

- Входы принадлежат к отделимым гиперплоскостью множествам; это значит, что существует такой вектор $\mathbf{u}$ (нормаль), что

$$\begin{aligned} \forall \mathbf{x} \in \mathcal{C}_1 \quad & \mathbf{u}^\top \mathbf{x} > 0, \\ \forall \mathbf{x} \in \mathcal{C}_2 \quad & \mathbf{u}^\top \mathbf{x} < 0. \end{aligned}$$

- Цель обучения — сделать так, чтобы веса перцептрона $\mathbf{w}$ образовывали такой вектор $\mathbf{u}$.

Доказательство

- Заменим $\mathcal{C}_2$ на $-\mathcal{C}_2 = \{-\mathbf{x} \mid \mathbf{x} \in \mathcal{C}_2\}$. Это позволит нам объединить два неравенства в одно:

$$\forall \mathbf{x} \in \mathcal{C} \ \mathbf{w}^\top \mathbf{x} > 0, \quad \mathcal{C} = \mathcal{C}_1 \cup (-\mathcal{C}_2).$$

- Итак, мы предполагаем, что входы принадлежат к отделимым гиперплоскостью множествам; это означает, что существует такой вектор $\mathbf{u}$ (нормаль к той самой гиперплоскости), что

$$\begin{aligned} \forall \mathbf{x} \in \mathcal{C}_1 \quad & \mathbf{u}^\top \mathbf{x} > 0, \\ \forall \mathbf{x} \in \mathcal{C}_2 \quad & \mathbf{u}^\top \mathbf{x} < 0. \end{aligned}$$

Цель обучения перцептрона — сделать так, чтобы веса перцептрона w образовывали такой вектор $\mathbf{u}$.

Доказательство

- Геометрически: хотим построить вектор $\mathbf{w}$, который образует острые углы со всеми тестовыми примерами $\mathbf{x}$.
- Обучение: если вдруг обнаружится вектор, с которым $\mathbf{w}$ образует тупой угол, мы просто прибавим его к $\mathbf{w}$ (с константой η, конечно).
- Осталось понять, почему этот процесс закончится. Для этого рассмотрим вектор, который *действительно* образует с ними тупые углы (он существует, это $\mathbf{u}$), и будем доказывать, что последовательность длин проекций $\mathbf{w}$ на этот вектор не может быть бесконечной.

Доказательство

- $\mathbf{w}^0, \mathbf{w}^1, \ldots$ — веса перцептрона по мере обучения, $\mathbf{x}^0, \mathbf{x}^1, \ldots$ — векторы тестовых примеров.
- W.l.o.g. $\mathbf{w}^0 = 0$, все тестовые примеры принадлежат $\mathcal{C}$, и на всех тестовых примерах $(\mathbf{w}^k)^\top \mathbf{x}^k < 0$, т.е. перцептрон не справляется с тестом (если он справляется с тестом, то веса перцептрона никак не меняются, поэтому такие тесты можно просто исключить из последовательности).
- Тогда правило обучения выглядит как

$$\mathbf{w}_i^{k+1} = \mathbf{w}_i^k + \eta \mathbf{x}_i^k,$$

и в наших предположениях

$$\mathbf{w}^k = \eta \sum_{j=0}^{k-1} \mathbf{x}^j.$$

Доказательство

- Идея: получить две серии противоположных оценок на длину векторов $\|\mathbf{w}^i\|$ и показать, что они не могут обе иметь место до бесконечности.
- Это и будет означать, что любая конечная последовательность тестов из двух фиксированных линейно отделимых множеств рано или поздно закончится, т.е. все последующие тесты перцептрон будет проходить успешно.

Доказательство

- Рассмотрим решение $\mathbf{u}$ и обозначим через α минимальную проекцию любого из $\mathbf{x}^j$ на $\mathbf{u}$:

$$\alpha = \min_j \mathbf{u}^\top \mathbf{x}^j$$

(поскольку разных тестов конечное число, $\alpha > 0$).

- Тогда

$$\mathbf{u}^\top \mathbf{w}^{k+1} = \eta \mathbf{u}^\top \sum_{j=0}^{k} \mathbf{x}^j \geq \eta \alpha k.$$

Доказательство

- $\mathbf{u}^\top \mathbf{w}^{k+1} = \eta \mathbf{u}^\top \sum_{j=0}^{k} \mathbf{x}^j \geq \eta \alpha k.$
- Вспомним неравенство Коши–Буняковского:

$$\|\mathbf{x}\|^2 \|\mathbf{y}\|^2 \geq (\mathbf{x}^\top \mathbf{y})^2.$$

- Применительно к нашей задаче:

$$\|\mathbf{u}\|^2 \|\mathbf{w}^{k+1}\|^2 \geq (\eta \alpha k)^2, \quad \|\mathbf{w}^{k+1}\|^2 \geq \frac{(\eta \alpha k)^2}{\|\mathbf{u}\|^2}.$$

Доказательство

- $\|\mathbf{u}\|^2\|\mathbf{w}^{k+1}\|^2 \geq (\eta\alpha k)^2, \quad \|\mathbf{w}^{k+1}\|^2 \geq \frac{(\eta\alpha k)^2}{\|\mathbf{u}\|^2}.$
- Т.е. на каждой итерации длина вектора $\mathbf{w}^k$ возрастает линейно.

Доказательство

- С другой стороны, $\mathbf{w}^{k+1} = \mathbf{w}^k + \eta x^k$. Поскольку $(\mathbf{w}^k)^\top x^k < 0$,

$$\|\mathbf{w}^{k+1}\|^2 = \|\mathbf{w}^k\|^2 + 2\eta(\mathbf{w}^k)^\top \mathbf{x}^k + \eta^2\|\mathbf{x}^k\|^2 \leq \|\mathbf{w}^k\|^2 + \eta^2\|\mathbf{x}^k\|^2.$$

- Следовательно, $\|\mathbf{w}^{k+1}\|^2 - \|\mathbf{w}^k\|^2 \leq \eta^2\|\mathbf{x}^k\|^2$.
- Просуммируем по k:

$$\|\mathbf{w}^{k+1}\|^2 \leq \eta^2 \sum_{j=0}^{k} \|\mathbf{x}^k\|^2 \leq \eta^2 \beta k,$$

где $\beta = \max_j \|\mathbf{x}^j\|^2$.

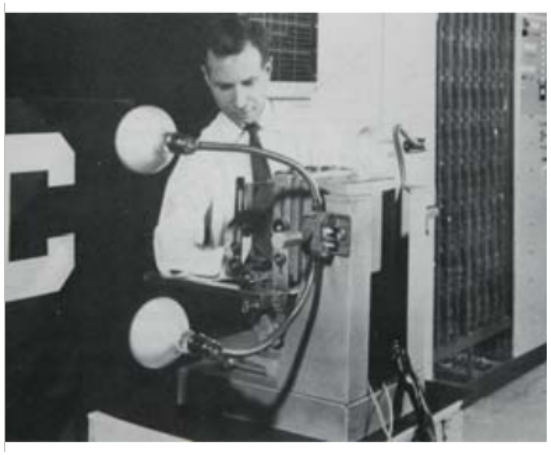
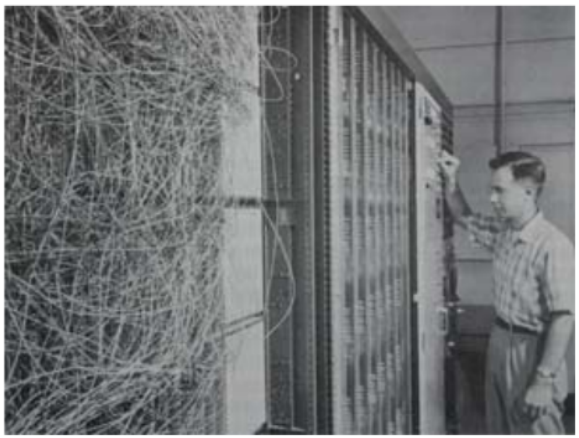
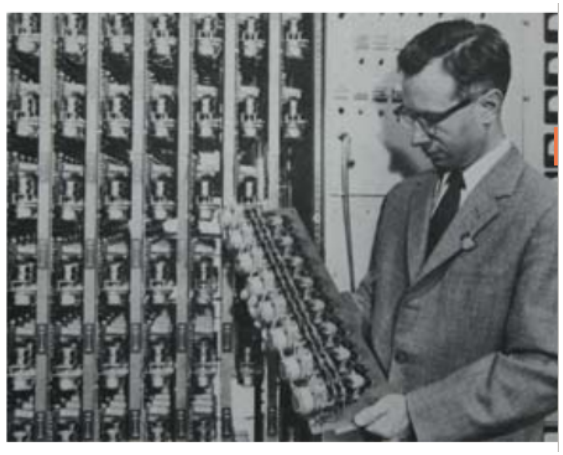
Доказательство

- Итак, получились две оценки:

$$\frac{(\eta\alpha)^2}{\|\mathbf{u}\|^2}k^2 \leq \|\mathbf{w}^{k+1}\|^2 \leq \eta^2\beta k.$$

- Рано или поздно с ростом k эти оценки войдут в противоречие друг с другом.
- Это и значит, что последовательность применения одних и тех же тестовых примеров не может быть бесконечной.

Обучение перцептрона в Mark 1

Центр Речевых Технологий

Outline

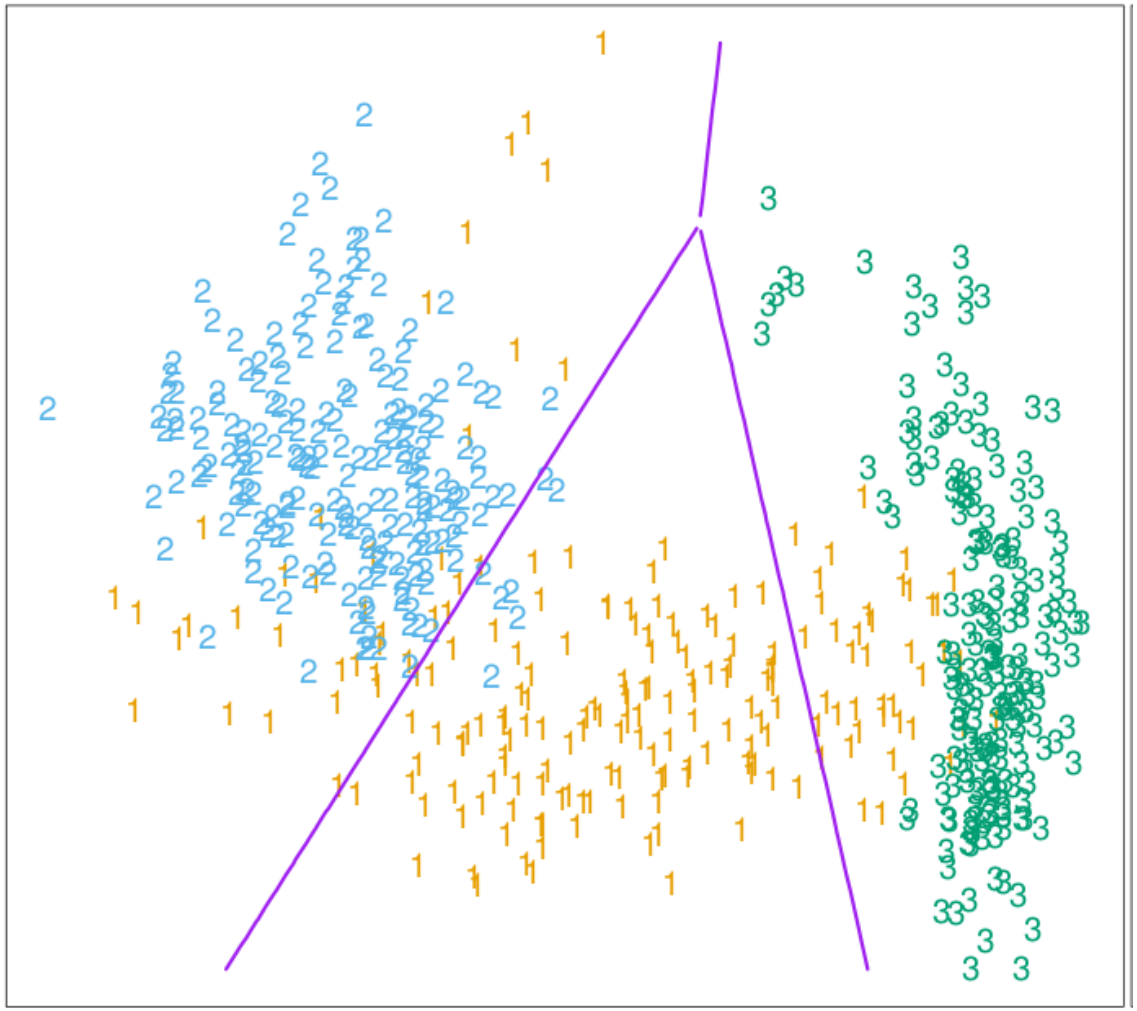
В прошлый раз

- В прошлый раз мы рассмотрели задачу классификации.
- Построили разделяющую гиперплоскость методом наименьших квадратов.
- И методом линейного дискриминанта Фишера.
- А потом научились обучать перцептрон и доказали сходимость метода.

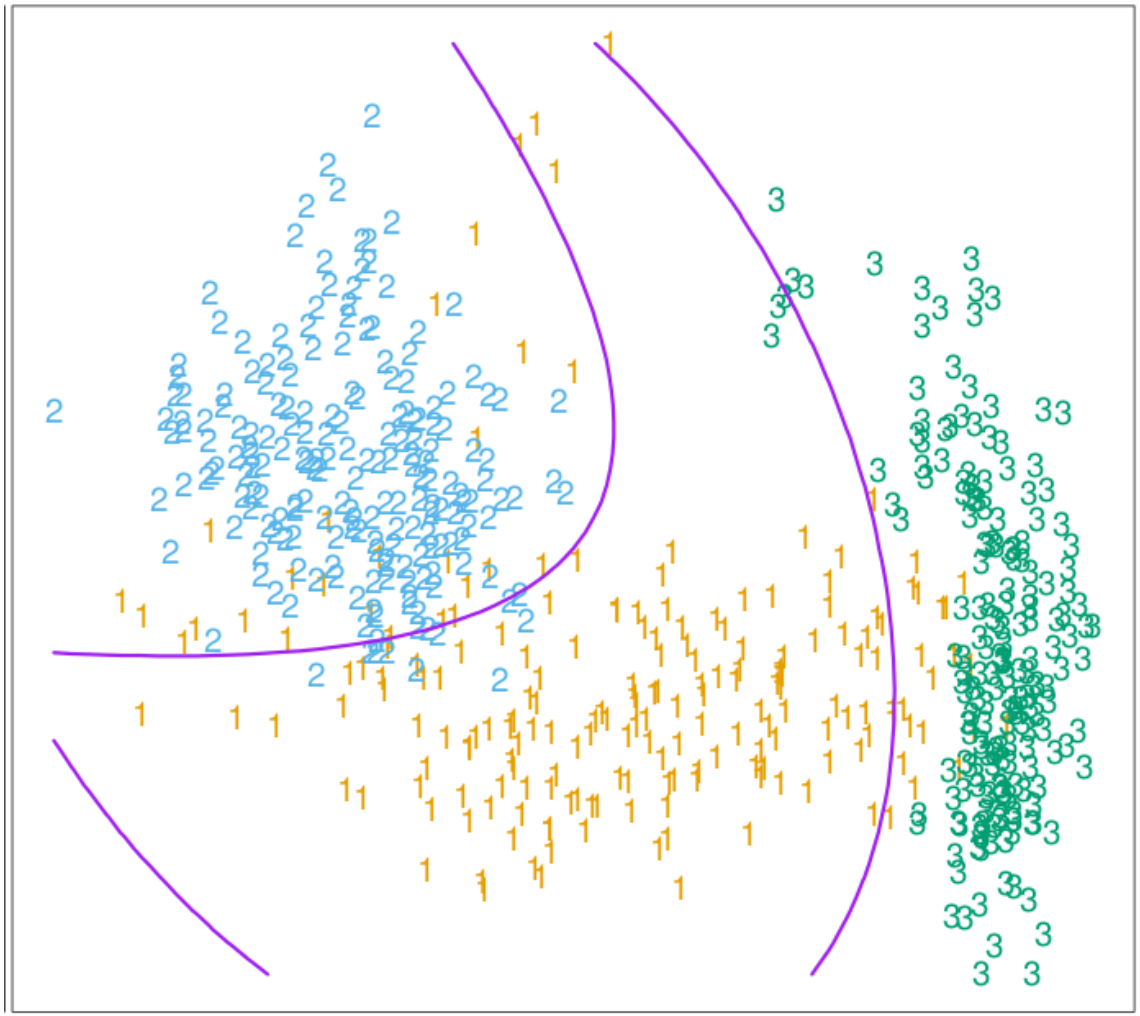
Нелинейные поверхности

- Мы учились проводить разделяющие гиперплоскости.
- Но как же нелинейные поверхности?
- Можно делать нелинейные из линейных, увеличивая размерность.

Нелинейные поверхности



Нелинейные поверхности



Генеративные модели

- Теперь классификация через генеративные модели: давайте каждому классу сопоставим плотность $p(\mathbf{x} \mid \mathcal{C}_k)$, найдём априорные распределения $p(\mathcal{C}_k)$, будем искать $p(\mathcal{C}_k \mid \mathbf{x})$ по теореме Байеса.
- Для двух классов:

$$p(\mathcal{C}_1 \mid \mathbf{x}) = \frac{p(\mathbf{x} \mid \mathcal{C}_1)p(\mathcal{C}_1)}{p(\mathbf{x} \mid \mathcal{C}_1)p(\mathcal{C}_1) + p(\mathbf{x} \mid \mathcal{C}_2)p(\mathcal{C}_2)}.$$

Генеративные модели

- Перепишем:

$$p(\mathcal{C}_1 \mid \mathbf{x}) = \frac{p(\mathbf{x} \mid \mathcal{C}_1)p(\mathcal{C}_1)}{p(\mathbf{x} \mid \mathcal{C}_1)p(\mathcal{C}_1) + p(\mathbf{x} \mid \mathcal{C}_2)p(\mathcal{C}_2)} = \frac{1}{1 + e^{-a}} = \sigma(a),$$

где

$$a = \ln \frac{p(\mathbf{x} \mid \mathcal{C}_1)p(\mathcal{C}_1)}{p(\mathbf{x} \mid \mathcal{C}_2)p(\mathcal{C}_2)}, \qquad \sigma(a) = \frac{1}{1 + e^{-a}}.$$

Генеративные модели

- $\sigma(a)$ – *логистический сигмоид*:

$$\sigma(a) = \frac{1}{1 + e^{-a}}$$

- $\sigma(-a) = 1 - \sigma(a)$.
- $a = \ln\left(\frac{\sigma}{1-\sigma}\right)$ – *логит-функция*.

Упражнение. Докажите эти свойства.

Несколько классов

- В случае нескольких классов получится

$$p(\mathcal{C}_k \mid \mathbf{x}) = \frac{p(\mathbf{x} \mid \mathcal{C}_k)p(\mathcal{C}_k)}{\sum_j p(\mathbf{x} \mid \mathcal{C}_j)p(\mathcal{C}_j)} = \frac{e^{a_k}}{\sum_j e^{a_j}}.$$

- Здесь $a_k = \ln p(\mathbf{x} \mid \mathcal{C}_k)p(\mathcal{C}_k)$.
- $\frac{e^{a_k}}{\sum_j e^{a_j}}$ – нормализованная экспонента, или softmax-функция (сглаженный максимум).

Пример

- Давайте рассмотрим гауссовы распределения для классов:

$$p(\mathbf{x} \mid \mathcal{C}_k) = \mathcal{N}(\mathbf{x} \mid \boldsymbol{\mu}_k, \Sigma).$$

- Сначала пусть Σ у всех одинаковые, а классов всего два.
- Посчитаем логистический сигмоид...

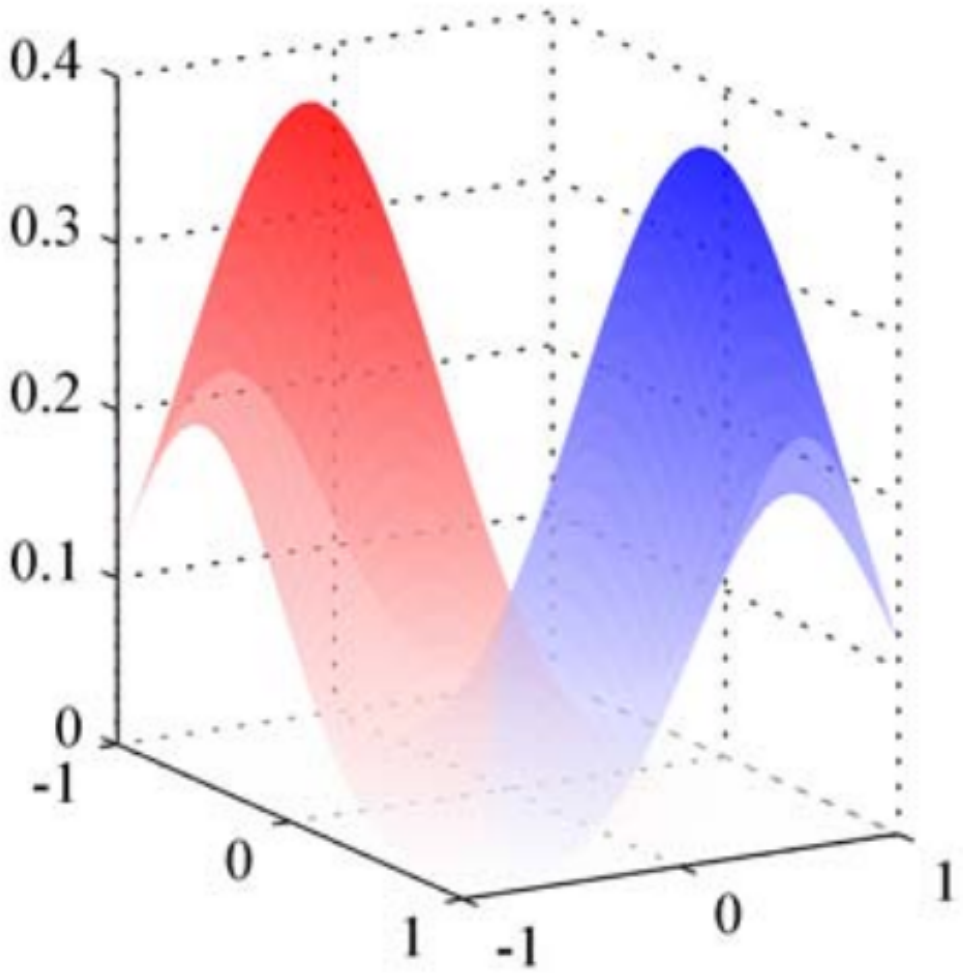
Пример

- ...получится

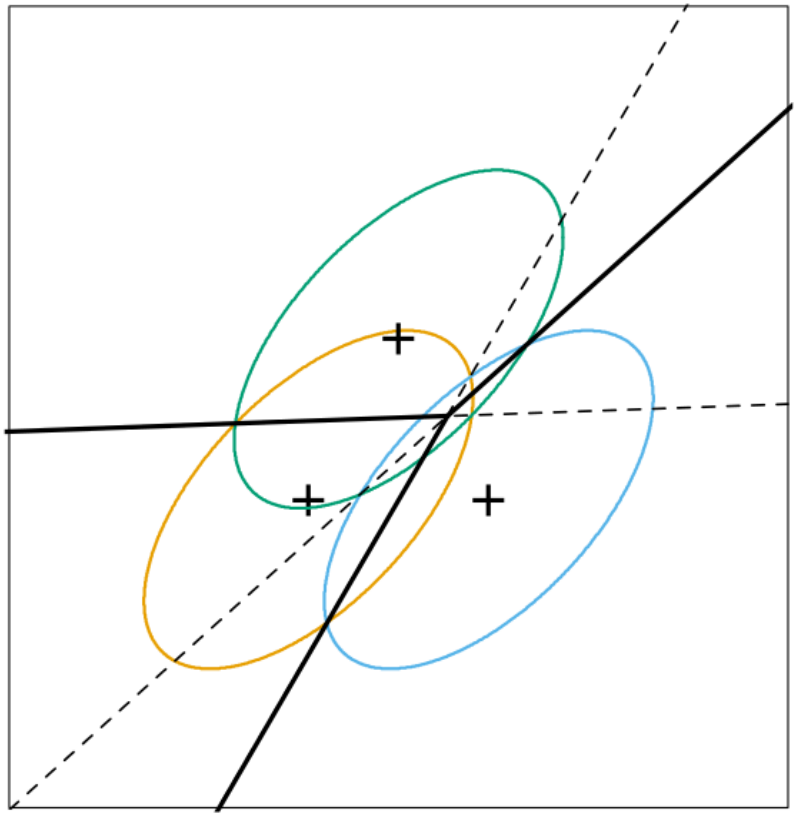
$$p(\mathcal{C}_1 \mid \mathbf{x}) = \sigma(\mathbf{w}^\top \mathbf{x} + w_0), \text{ где}$$
$$\mathbf{w} = \Sigma^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2),$$
$$w_0 = -\frac{1}{2}\boldsymbol{\mu}_1^\top \Sigma^{-1} \boldsymbol{\mu}_1 + \frac{1}{2}\boldsymbol{\mu}_2^\top \Sigma^{-1} \boldsymbol{\mu}_2 + \ln \frac{p(\mathcal{C}_1)}{p(\mathcal{C}_2)}.$$

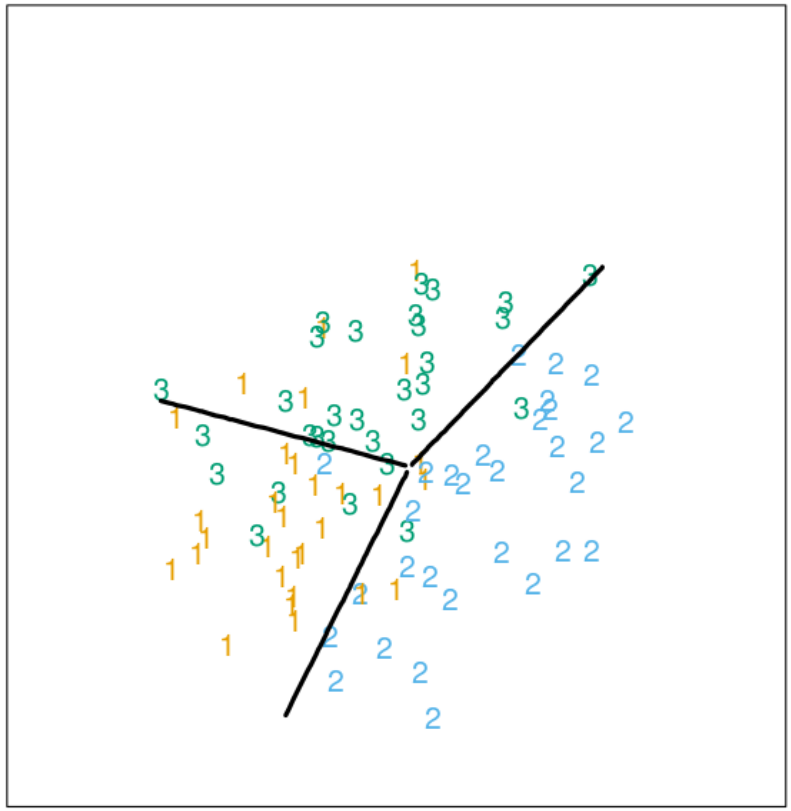
- Т.е. в аргументе сигмоида получается линейная функция от $\mathbf{x}$. Поверхности уровня – это когда $p(\mathcal{C}_1 \mid \mathbf{x})$ постоянно, т.е. гиперплоскости в пространстве $\mathbf{x}$. Априорные вероятности $p(\mathcal{C}_k)$ просто сдвигают эти гиперплоскости.

Разделяющая гиперплоскость



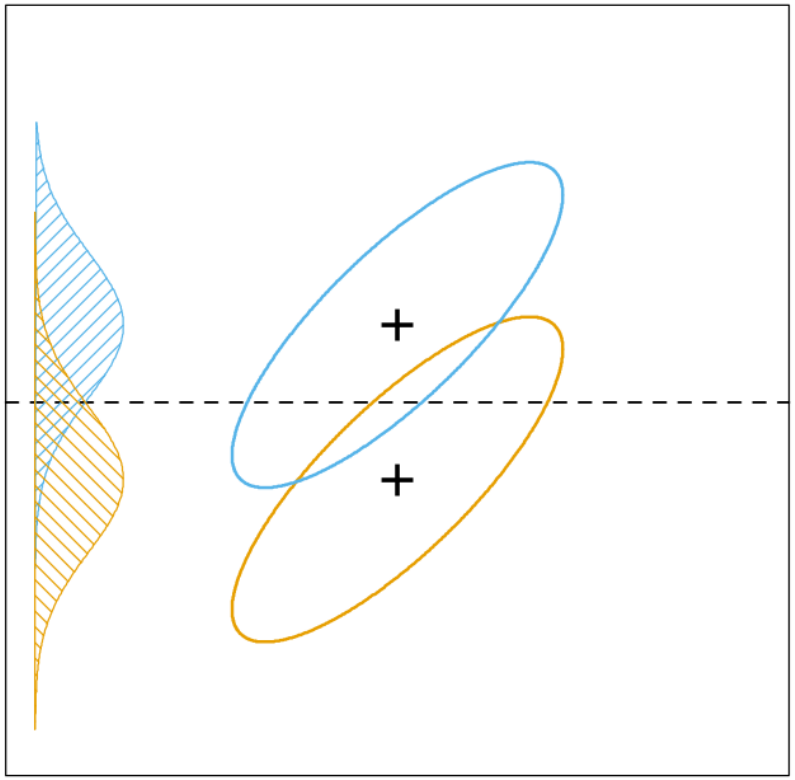
Разделяющая гиперплоскость

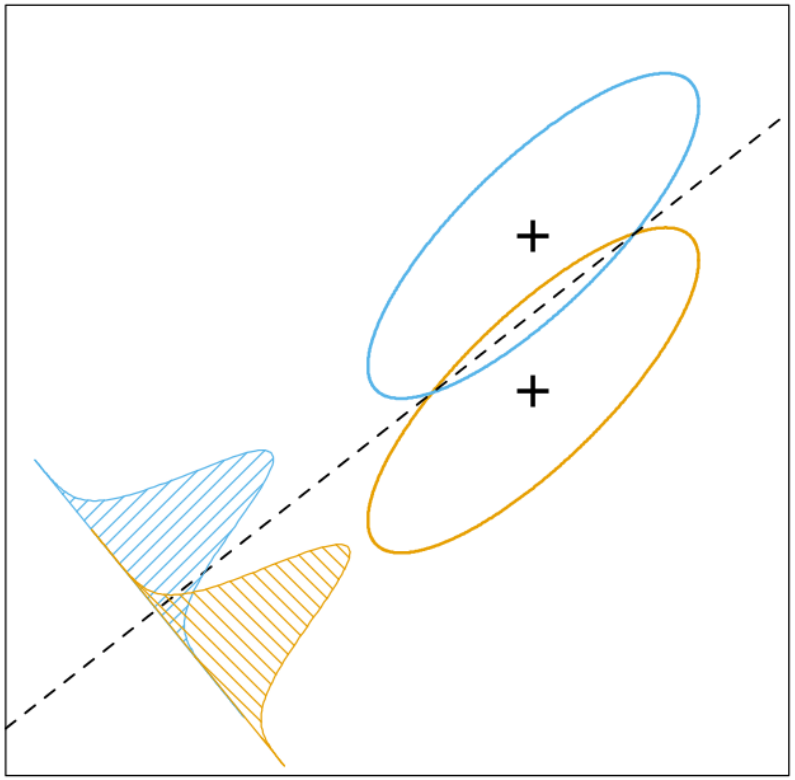




Дискриминант Фишера

Кстати, с дискриминантом Фишера эта разделяющая поверхность отлично сходится.





Несколько классов

- С несколькими классами получится тоже примерно так же:

$$\delta_k(\mathbf{x}) = \mathbf{x}^\top \Sigma^{-1} \boldsymbol{\mu}_k - \frac{1}{2} \boldsymbol{\mu}_k^\top \Sigma^{-1} \boldsymbol{\mu}_k + \ln \pi_k,$$

  где $\pi_k = p(\mathcal{C}_k)$.
- Получились линейные $\delta_k(\mathbf{x})$, и опять разделяющие поверхности линейные (тут разделяющие поверхности – когда две максимальных вероятности равны).
- Этот метод называется LDA – linear discriminant analysis.

Метод максимального правдоподобия

- Как оценить распределения $p(\mathbf{x} \mid \mathcal{C}_k)$, если даны только данные?
- Можно по методу максимального правдоподобия.
- Опять рассмотрим тот же пример: два класса, гауссианы с одинаковой матрицей ковариаций, и есть $D = \{\mathbf{x}_n, t_n\}_{n=1}^N$, где $t_n = 1$ значит $\mathcal{C}_1$, $t_n = 0$ значит $\mathcal{C}_2$.
- Обозначим $p(\mathcal{C}_1) = \pi$, $p(\mathcal{C}_2) = 1 - \pi$.

Метод максимального правдоподобия

- Для одной точки в классе $\mathcal{C}_1$:

$$p(\mathbf{x}_n, \mathcal{C}_1) = p(\mathcal{C}_1)p(\mathbf{x}_n \mid \mathcal{C}_1) = \pi\mathcal{N}(\mathbf{x}_n \mid \mu_1, \Sigma).$$

- В классе $\mathcal{C}_2$:

$$p(\mathbf{x}_n, \mathcal{C}_2) = p(\mathcal{C}_2)p(\mathbf{x}_n \mid \mathcal{C}_2) = (1-\pi)\mathcal{N}(\mathbf{x}_n \mid \mu_2, \Sigma).$$

- Функция правдоподобия:

$$p(\mathbf{t} \mid \pi, \mu_1, \mu_2, \Sigma) = \\ = \prod_{n=1}^{N} \left[\pi\mathcal{N}(\mathbf{x}_n \mid \mu_1, \Sigma)\right]^{t_n} \left[(1-\pi)\mathcal{N}(\mathbf{x}_n \mid \mu_2, \Sigma)\right]^{1-t_n}.$$

Метод максимального правдоподобия

- Максимизируем логарифм правдоподобия. Сначала по π, там останется только

$$\sum_{n=1}^{N} \left[t_n \ln \pi + (1 - t_n) \ln(1 - \pi) \right],$$

и, взяв производную, получим, совершенно неожиданно,

$$\hat{\pi} = \frac{N_1}{N_1 + N_2}.$$

Метод максимального правдоподобия

- Теперь по μ_1; всё, что зависит от μ_1:

$$\sum_n t_n \ln \mathcal{N}(\mathbf{x}_n \mid \mu_1, \Sigma) = -\frac{1}{2} \sum_n t_n \left(\mathbf{x}_n - \boldsymbol{\mu}_1\right)^\top \Sigma^{-1} \left(\mathbf{x}_n - \boldsymbol{\mu}_1\right) + C.$$

- Берём производную, и получается, опять внезапно,

$$\hat{\mu}_1 = \frac{1}{N_1} \sum_{n=1}^{N} t_n \mathbf{x}_n.$$

- Аналогично,

$$\hat{\mu}_2 = \frac{1}{N_2} \sum_{n=1}^{N} (1 - t_n) \mathbf{x}_n.$$

Метод максимального правдоподобия

- Для матрицы ковариаций придётся постараться; в результате получится

$$\hat{\Sigma} = \frac{N_1}{N_1 + N_2}\mathbf{S}_1 + \frac{N_2}{N_1 + N_2}\mathbf{S}_2, \text{ где}$$

$$\mathbf{S}_1 = \frac{1}{N_1}\sum_{n \in \mathcal{C}_1} (\mathbf{x}_n - \boldsymbol{\mu}_1)(\mathbf{x}_n - \boldsymbol{\mu}_1)^\top,$$

$$\mathbf{S}_2 = \frac{1}{N_2}\sum_{n \in \mathcal{C}_2} (\mathbf{x}_n - \boldsymbol{\mu}_2)(\mathbf{x}_n - \boldsymbol{\mu}_2)^\top.$$

- Тоже совершенно неожиданно: взвешенное среднее оценок для двух матриц ковариаций.

Метод максимального правдоподобия

- Это самым прямым образом обобщается на случай нескольких классов.

Упражнение. Сделайте это.

Разные матрицы ковариаций

- А вот с разными матрицами ковариаций уже будет по-другому.
- Квадратичные члены не сократятся.
- Разделяющие поверхности станут квадратичными; QDA – quadratic discriminant analysis.

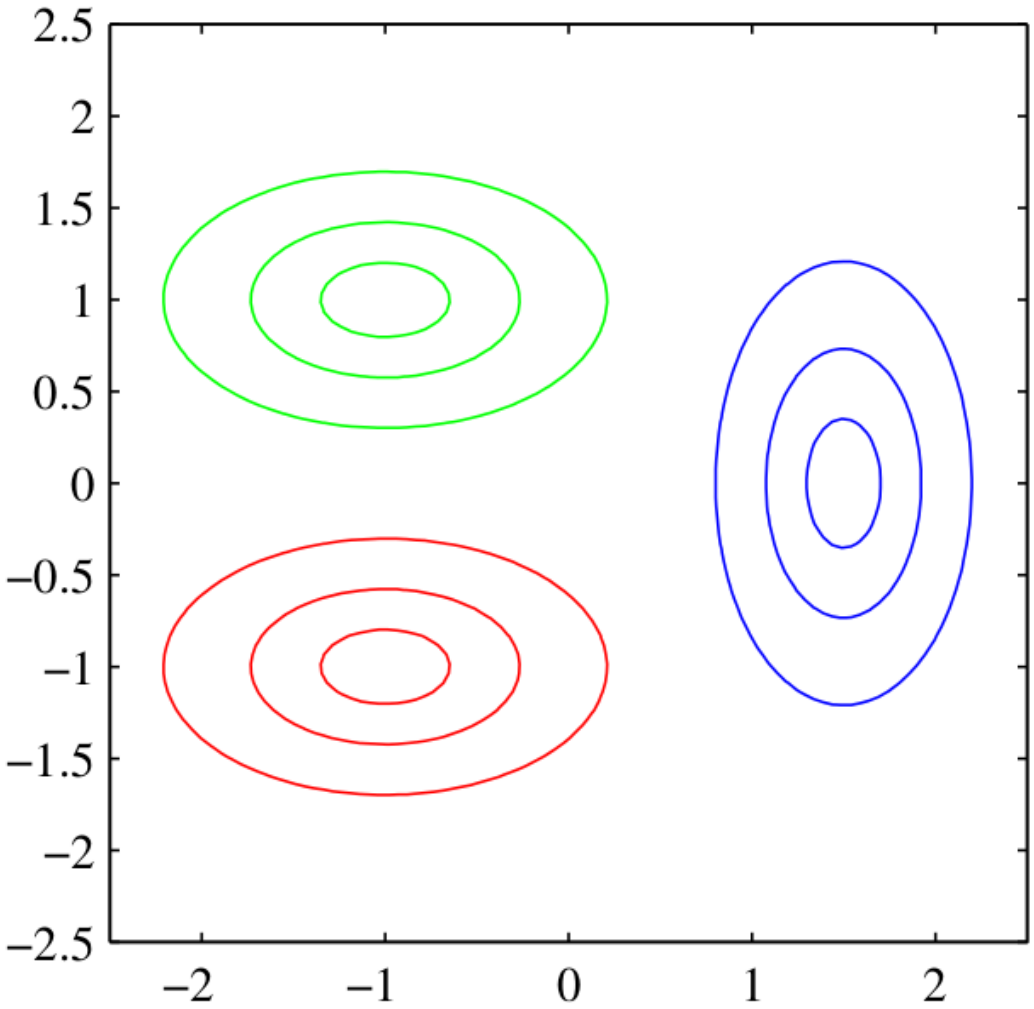
Разные матрицы ковариаций

- В QDA получится

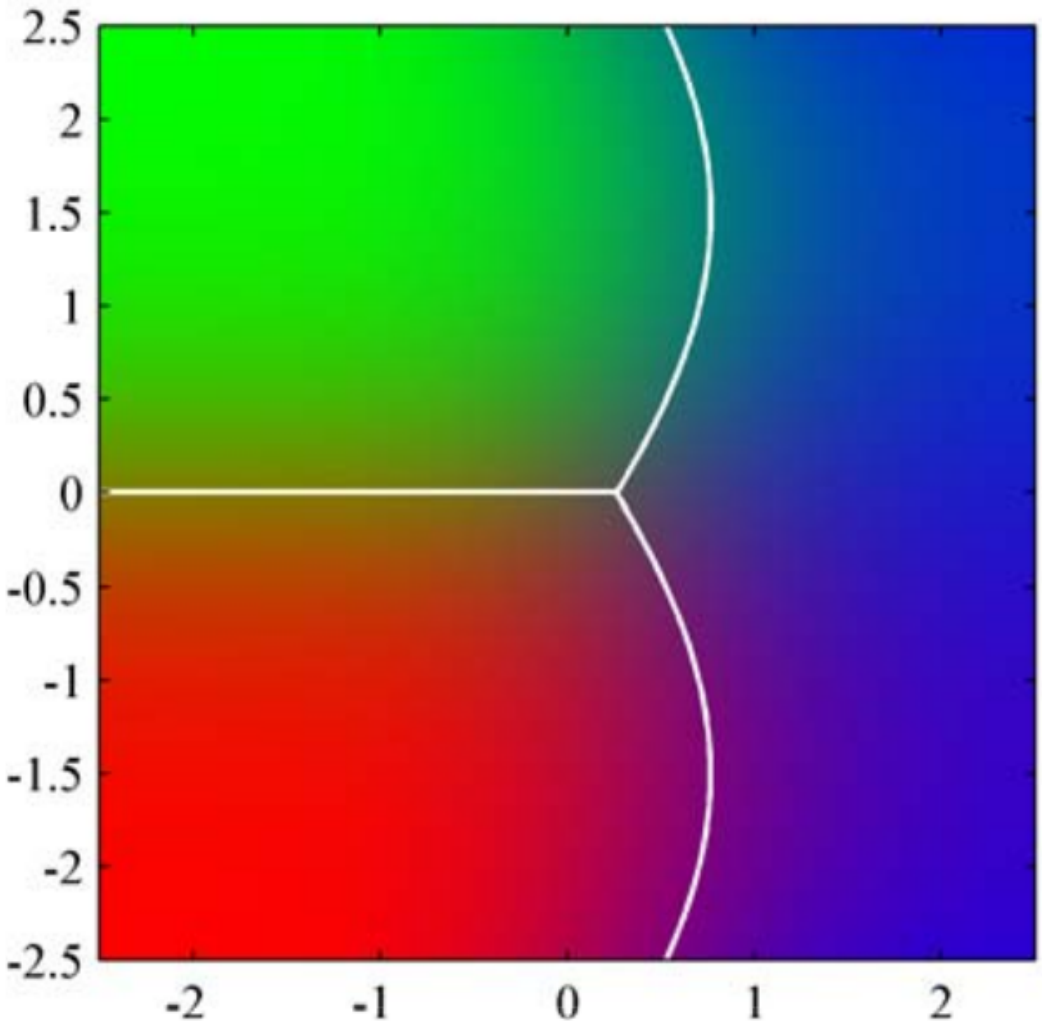
$$\delta_k(\mathbf{x}) = -\frac{1}{2}\log|\Sigma_k| - \frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_k)^{-1}\Sigma_k^{-1}(\mathbf{x}-\boldsymbol{\mu}_k) + \log\pi_k.$$

- Разделяющая поверхность между $\mathcal{C}_i$ и $\mathcal{C}_j$ – это $\{\mathbf{x} \mid \delta_i(\mathbf{x}) = \delta_j(\mathbf{x})\}$.
- Оценки максимального правдоподобия такие же, только надо отдельно матрицы ковариаций оценивать.

Разные матрицы ковариаций

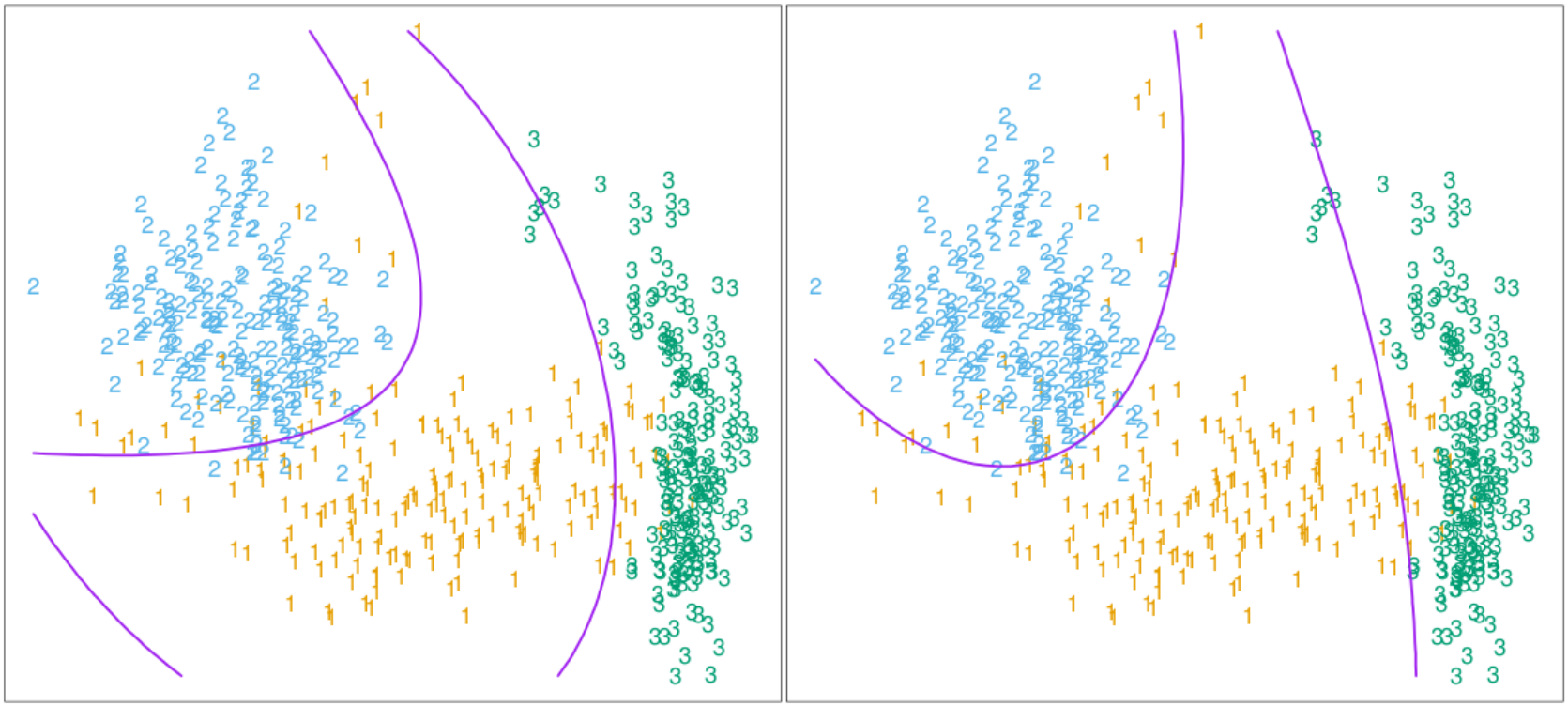


Разные матрицы ковариаций



LDA vs. QDA

Разница между LDA с квадратичными членами и QDA обычно невелика.



LDA vs. QDA

- LDA и QDA очень хорошо работают на практике. Всегда первая идея в классификации.
- Число параметров:
  - у LDA $(K-1)(d+1)$ параметр: по $d+1$ на каждую разницу вида $\delta_k(\mathbf{x}) - \delta_K(\mathbf{x})$;
  - у QDA $(K-1)(d(d+3)/2+1)$ параметр, но он выглядит гораздо лучше своих лет.

LDA vs. QDA

- Почему хорошо работают?
- Скорее всего, потому, что линейные и квадратичные оценки достаточно стабильны: даже если bias относительно большой (как будет, если данные всё-таки не гауссианами порождены), variance будет маленькой.

RDA

- Компромисс между LDA и QDA – регуляризованный дискриминантный анализ, RDA.
- Стянем ковариации каждого класса к общей матрице ковариаций:

$$\hat{\Sigma}_k(\alpha) = \alpha\hat{\Sigma}_k + (1-\alpha)\hat{\Sigma},$$

где $\hat{\Sigma}_k$ – оценка из QDA, $\hat{\Sigma}$ – оценка из LDA.

- Или стянем к единичной матрице:

$$\hat{\Sigma}_k(\gamma) = \gamma\hat{\Sigma}_k + (1-\gamma)\hat{\sigma}^2\mathbf{I}.$$

Снижение ранга в LDA

- Предположим, что размерность d больше, чем число классов K.
- Тогда центроиды классов $\hat{\mu}_k$ лежат в подпространстве размерности $\leq K - 1$.
- И когда мы определяем ближайший центроид, нам достаточно считать расстояния только в этом подпространстве.
- Таким образом, можно сократить ранг задачи.

Снижение ранга в LDA

- Куда именно проецировать? Не обязательно само подпространство, порождённое центроидами, будет оптимальным.
- Это мы уже проходили: для размерности 1 это линейный дискриминант Фишера.
- Это он и есть: оптимальное подпространство будет там, где межклассовая дисперсия максимальна по отношению к внутриклассовой.

Thank you!

Спасибо за внимание!