

Машинное обучение

Лекция 3. Вероятностный взгляд на машинное обучение



Упражнение с прошлой лекции: поиск ошибок

```
from random import randint

def loss(x, answer):
    return max([0, 1 - answer * f(x)])

def der_loss(x, answer):
    return -1.0 if 1 - answer * f(x) > 0 else 0.0

def fit(X_train, y_train):
    for k in range(10000):
        rand_index = randint(0, len(X_train))
        x = X_train[rand_index]
        y = y_train[rand_index]

        step = 0.01

        w -= x * step * y * der_loss(x, y)
```

Упражнение с прошлой лекции: поиск ошибок

```
from random import randint

def loss(x, answer):
    return max([0, 1 - answer * f(x)])

def der_loss(x, answer):
    return -1.0 if 1 - answer * f(x) > 0 else 0.0

def fit(X_train, y_train):
    for k in range(10000):
        rand_index = randint(0, len(X_train))
        x = X_train[rand_index]
        y = y_train[rand_index]

        step = 0.01

        w -= x * step * y * der_loss(x, y)
```

Ответы 0 и 1, а формулы – для +1 и -1

Упражнение с прошлой лекции: поиск ошибок

```
from random import randint

def loss(x, answer):
    return max([0, 1 - answer * f(x)])

def der_loss(x, answer):
    return -1.0 if 1 - answer * f(x) > 0 else 0.0

def fit(X_train, y_train):
    for k in range(10000):
        rand_index = randint(0, len(X_train))
        x = X_train[rand_index]
        y = y_train[rand_index]

        step = 0.01
        w -= x * step * y * der_loss(x, y)

    Ответы 0 и 1, а формулы – для +1 и -1

    Обновляем только w – надо либо обновлять
    и w0, либо добавлять фиктивный признак
```

Напоминание: часто используемые методы

- Линейные модели
- Решающие деревья
- Ансамбли решающих деревьев

Известные, но реже используемые модели

- Наивный Байесовский классификатор (NB)
- Линейный дискриминант Фишера (LDA)
- Квадратичный дискриминантный анализ (QDA, NDA)

Обсуждаем сегодня: вероятностный подход

- I. Байесовская классификация
- II. Восстановление плотности: «Наивный Байес», QDA и LDA
- III. Связь функций потерь и распределения данных
- IV. Оптимизация риска и анализ функций потерь

I. Байесовская классификация

Байесовский классификатор

По известному вектору признаков x алгоритм относит объект к классу $a(x)$ по правилу:

$$a(x) = \operatorname{argmax}_y P(y|x)$$

Байесовский классификатор

$$a(x) = \operatorname{argmax}_y P(y|x)$$



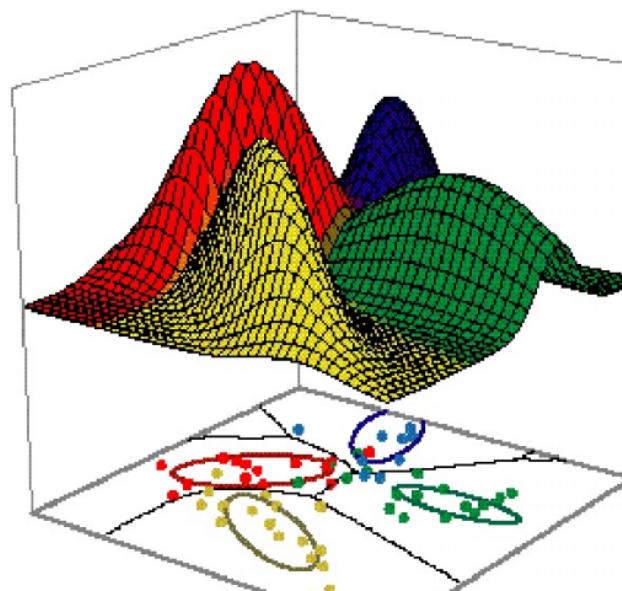
$$P(y|x) = \frac{P(x|y)P(y)}{P(x)}$$

$$a(x) = \operatorname{argmax}_y P(x|y)P(y)$$

Байесовский классификатор

$$a(x) = \operatorname{argmax}_y P(x|y)P(y)$$

Если $P(y)$ одинаковы для всех классов – мы просто выбираем класс, плотность которого больше в точке x :



Зачем нам понадобилась теорема Байеса

- $P(y|x)$ – вероятность класса y при признаках x
- X часто из вещественных чисел и признаков часто очень много
- Всевозможных значений признаков так много, что скорее всего каждый вектор x встретится только один или несколько раз
- Этого недостаточно для оценки $P(y|x)$

Что оценивается по обучающей выборке

- $P(x|y)$ – вероятность увидеть набор признаков x в классе y , если x дискретный
- Если координаты вектора x – вещественные, $P(x|y)$ – плотность распределения x
- Именно эту величину и можно оценивать по обучающей выборке
- А затем подставлять в классификатор:

$$a(x) = \operatorname{argmax}_y P(x|y)P(y)$$

Проблема нехватки данных

- Пример: в обучающей выборке 100 000 объектов с 10 000 признаков
- 100 000 точек в пространстве размерности 10 000 – очень мало
- Например, если x – бинарный, то y у него может быть 2^{10000} значений, что сильно больше 100 000
- Поэтому восстановить $P(x|y)$ как функцию от признаков x довольно трудно

Промежуточный итог

- Байесовский классификатор:

$$a(x) = \operatorname{argmax}_y P(x|y)P(y)$$

- **Обучение:** оценить по выборке $P(x|y)$ и $P(y)$
- Оценивать $P(x|y)$ как функцию **многих** переменных **затруднительно** – нужно много данных
- **Далее:** как преодолевать проблему нехватки данных и восстанавливать распределения по обучающей выборке

II. Восстановление плотности: «Наивный Байес», QDA и LDA

Проблема нехватки данных

- Пример: в обучающей выборке 100 000 объектов с 10 000 признаков
- 100 000 точек в пространстве размерности 10 000 – очень мало
- Например, если координаты вектора признаков x – бинарные (0 и 1), то $y|x$ может быть 2^{10000} значений, что сильно больше 100 000
- Поэтому восстановить $P(x|y)$ как функцию от вектора признаков x довольно трудно

Восстановление распределений

Оцениваем $P(y)$ и $P(x|y)$

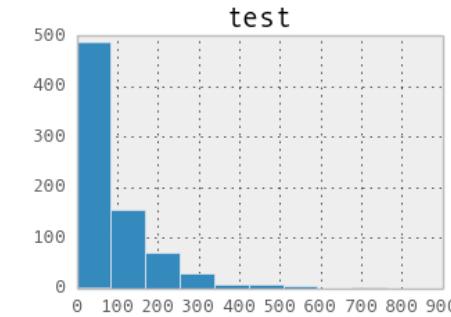
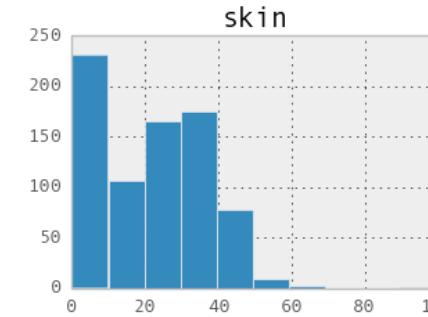
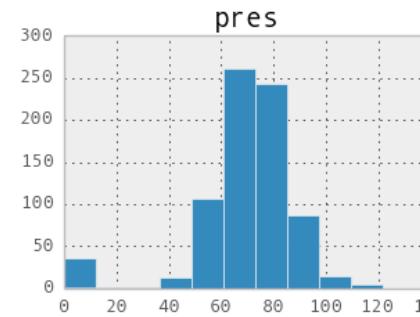
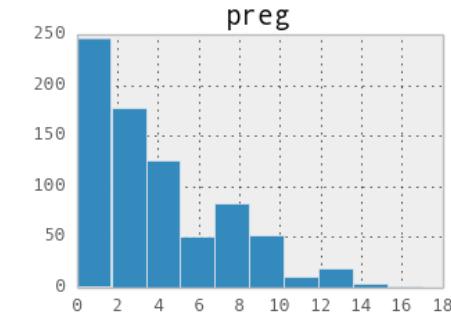
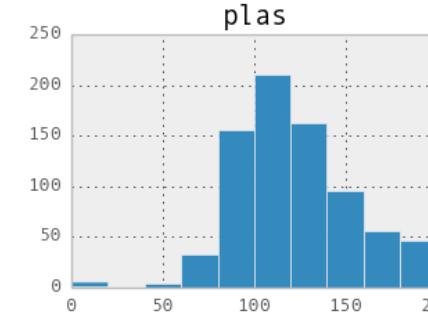
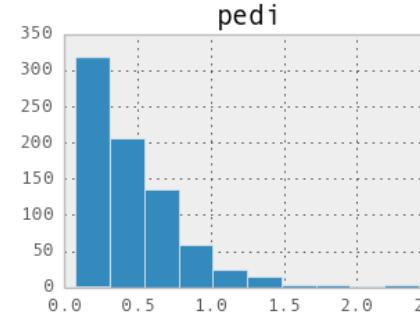
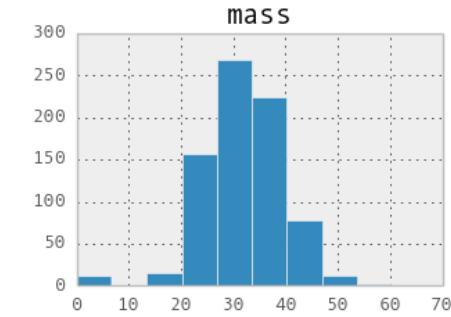
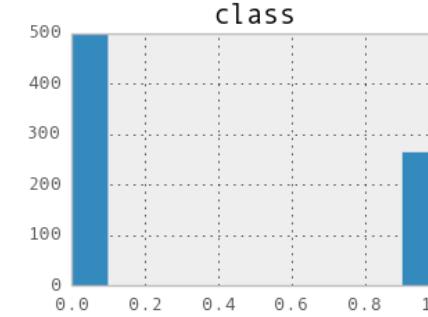
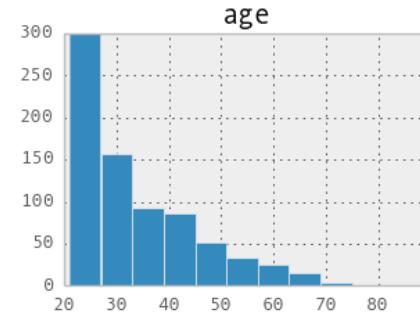
- $P(y)$ можно оценить как долю объектов класса y в обучающей выборке:

$$P(y) \approx \frac{l_y}{l}$$

Если соотношение классов в выборке воспроизводит реальное – это неплохая оценка.

- Как оценивать $P(x|y)$ – обсудим подробнее

Идея: распределения по каждому признаку



Решение 1 – «Наивный Байес»

Свести задачу восстановления $P(x|y)$ от оценки функции многих переменных к оценке функций одной переменной

Наивный байесовский классификатор

1. Байесовский классификатор:

$$a(x) = \operatorname{argmax}_y P(y|x) = \operatorname{argmax}_y P(x|y) P(y)$$

2. С «наивной» гипотезой:

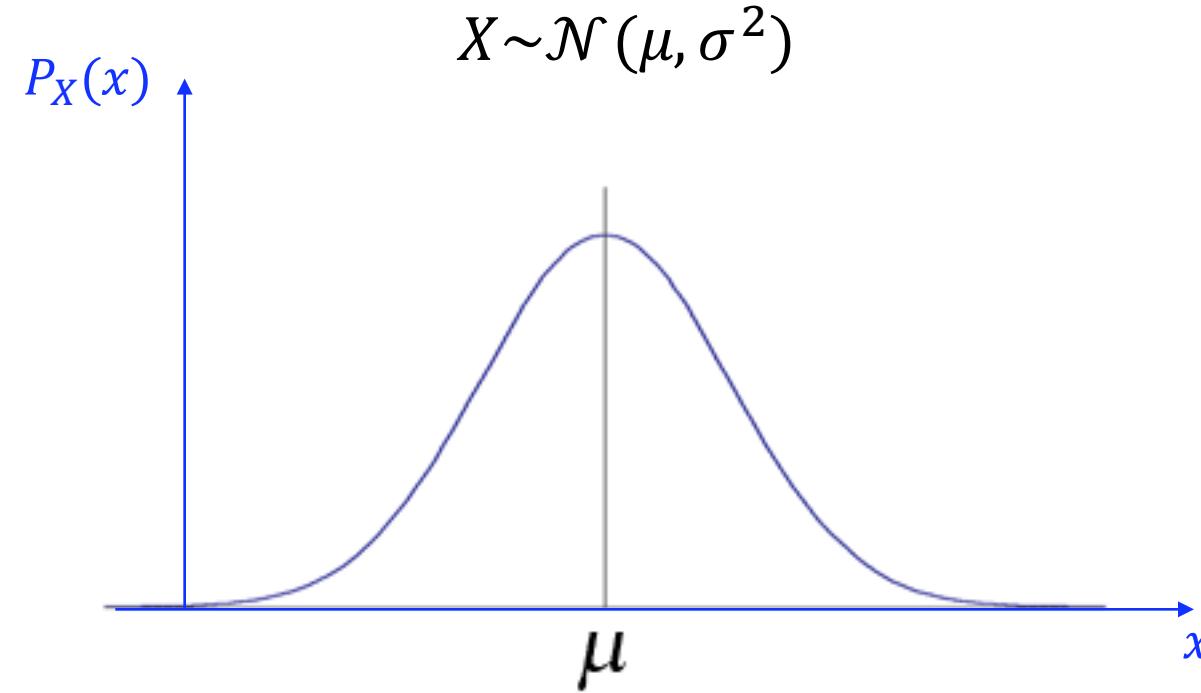
$$P(x|y) = P(x_{(1)}|y) P(x_{(2)}|y) \dots P(x_{(N)}|y)$$

$x_{(k)}$ – k-ый признак объекта x

Параметрическое восстановление распределений

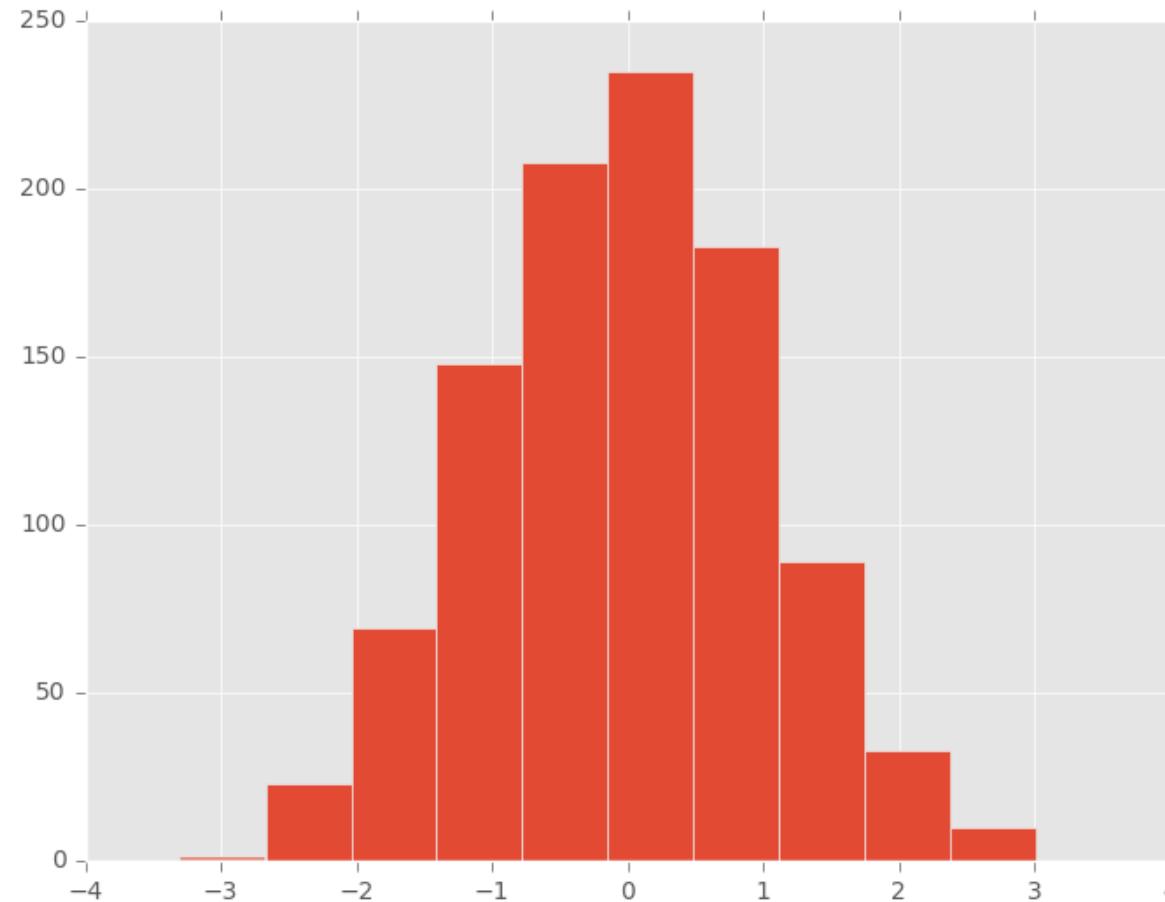
- Можем предположить, что распределение признаков похоже на какое-то стандартное – пуассоновское, экспоненциальное, нормальное
- И попробовать восстановить его (оценить параметры распределения)

Пример: нормальное распределение



$$p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Пример: нормальное распределение



Пример: нормальное распределение

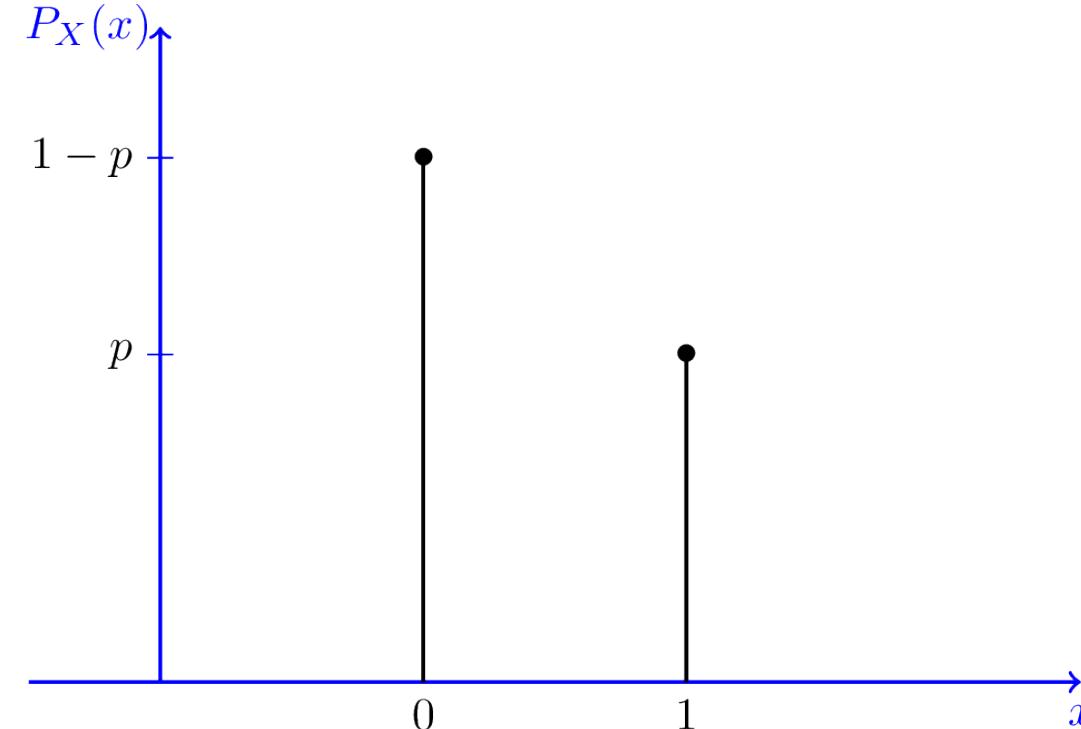
$$\hat{\mu} = \frac{1}{N} \sum_{i=1}^N x_i \quad \sigma^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \hat{\mu})^2$$

несмешенный вариант оценки для дисперсии:

$$\sigma^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \hat{\mu})^2$$

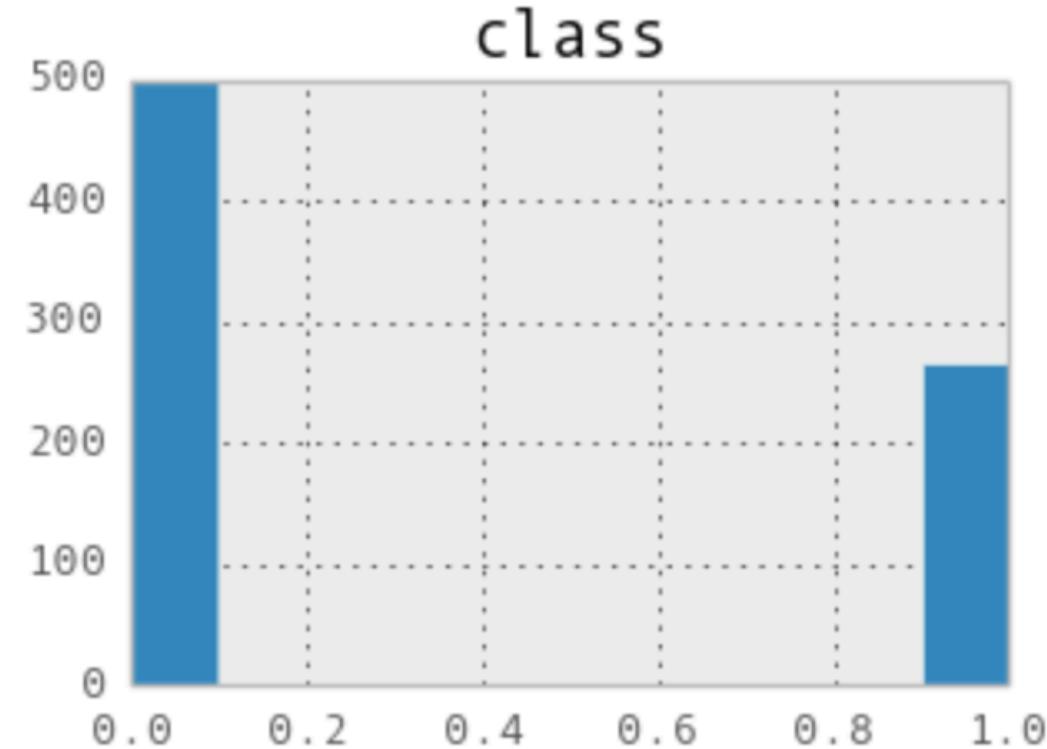
Другой пример: распределение Бернулли

$$X \sim Bernoulli(p)$$



$$\begin{aligned}P(x = 1) &= p \\P(x = 0) &= 1 - p\end{aligned}$$

Другой пример: распределение Бернулли



Другой пример: распределение Бернулли

Параметр p можно оценить долей случаев, в которых случайная величина равнялась 1:

$$\hat{p} = \frac{1}{N} \sum_{i=1}^N [x_i = 1]$$

Рекомендации по выбору распределения

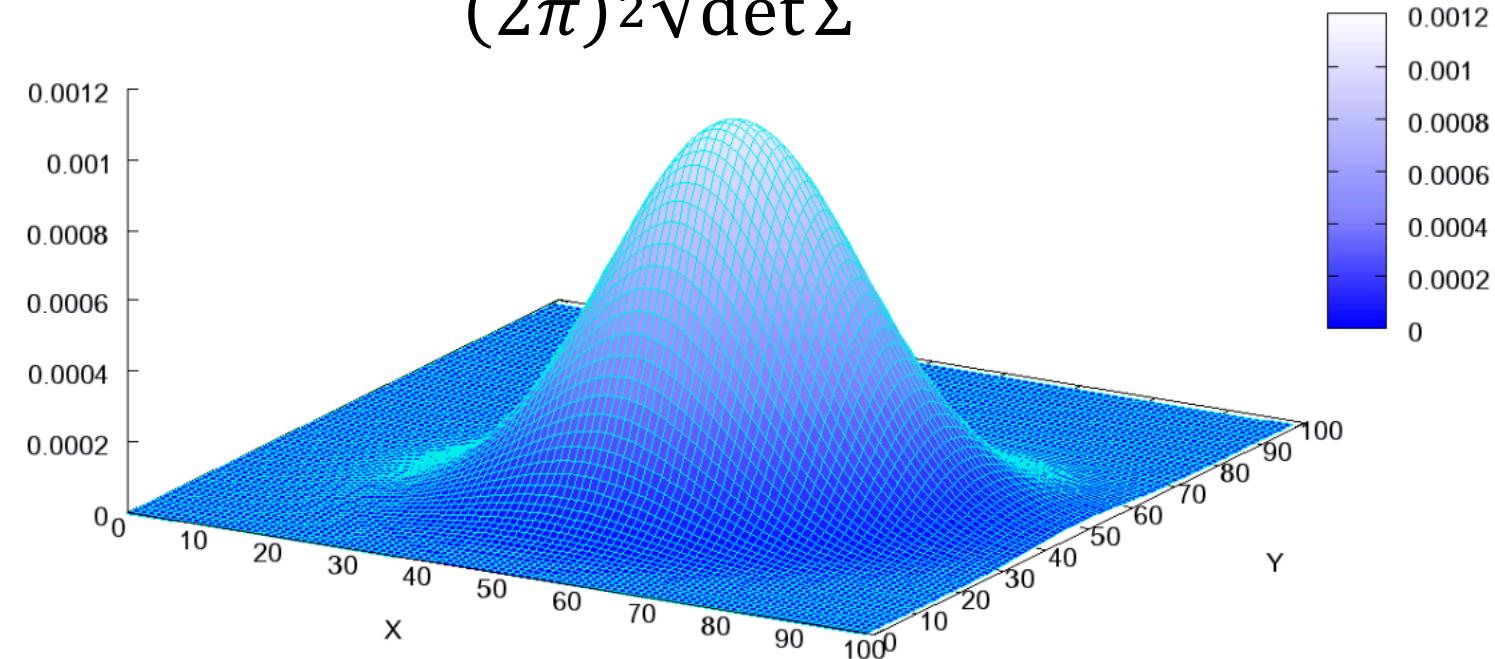
- Данные с разреженными дискретными признаками – **мультиномиальное** распределение
- Непрерывные признаки с маленьким разбросом – **нормальное** распределение
- Непрерывные признаки с выбросами в обучающей выборке – можно попробовать более «размазанные» распределения

Решение 2 – Параметрическая оценка многомерного распределения

- Применить для восстановления $P(x|y)$ параметрический подход, но восстанавливать сразу многомерное распределение
- Т.к. приближаем $P(x|y)$ распределением из некоторого узкого класса, число параметров может быть приемлемым

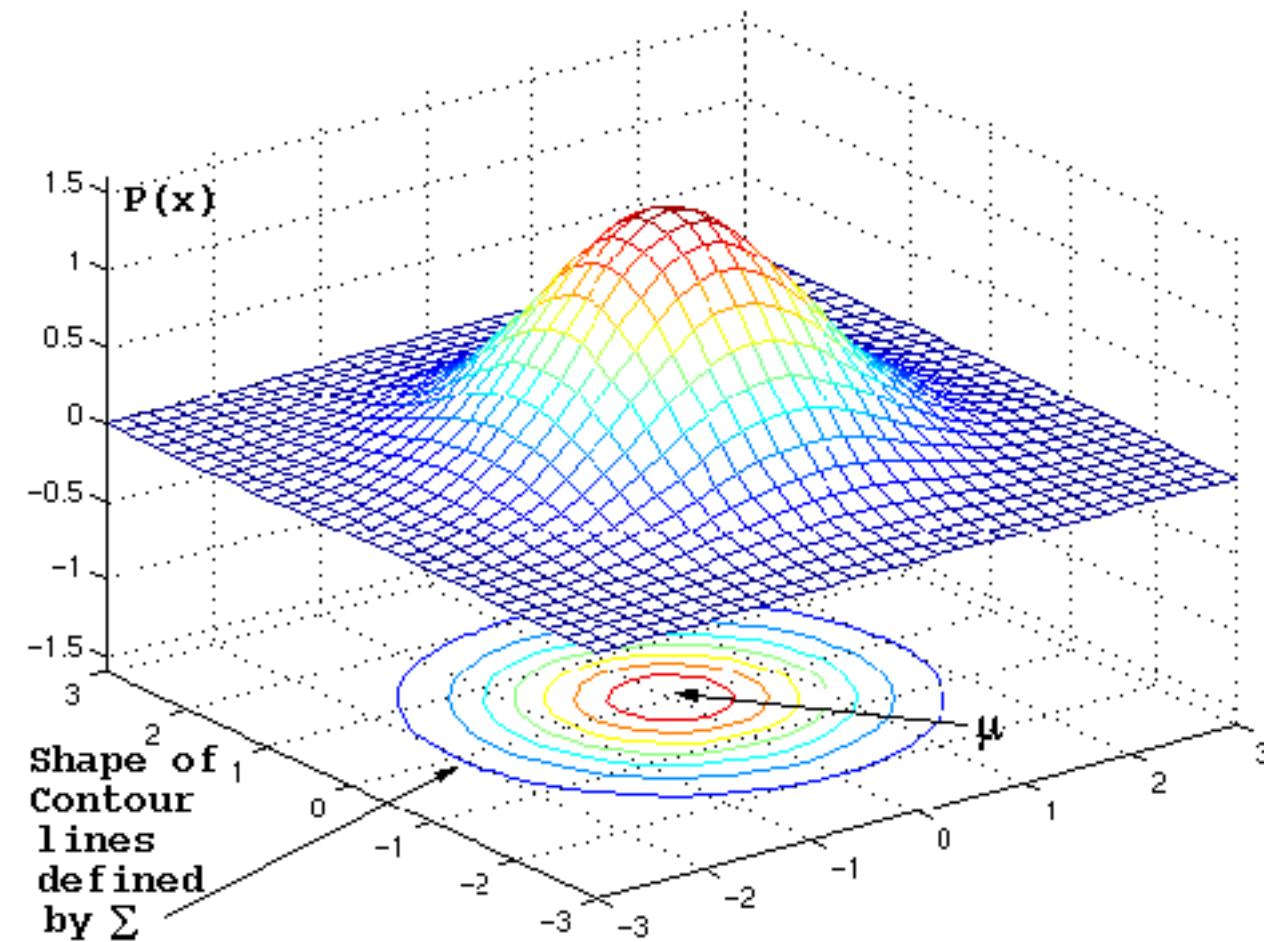
Пример: многомерное нормальное распределение

$$p(x) = \frac{1}{(2\pi)^{\frac{n}{2}} \sqrt{\det \Sigma}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)}$$



Параметры: вектор средних μ и матрица ковариаций Σ

Линии уровня плотности распределения



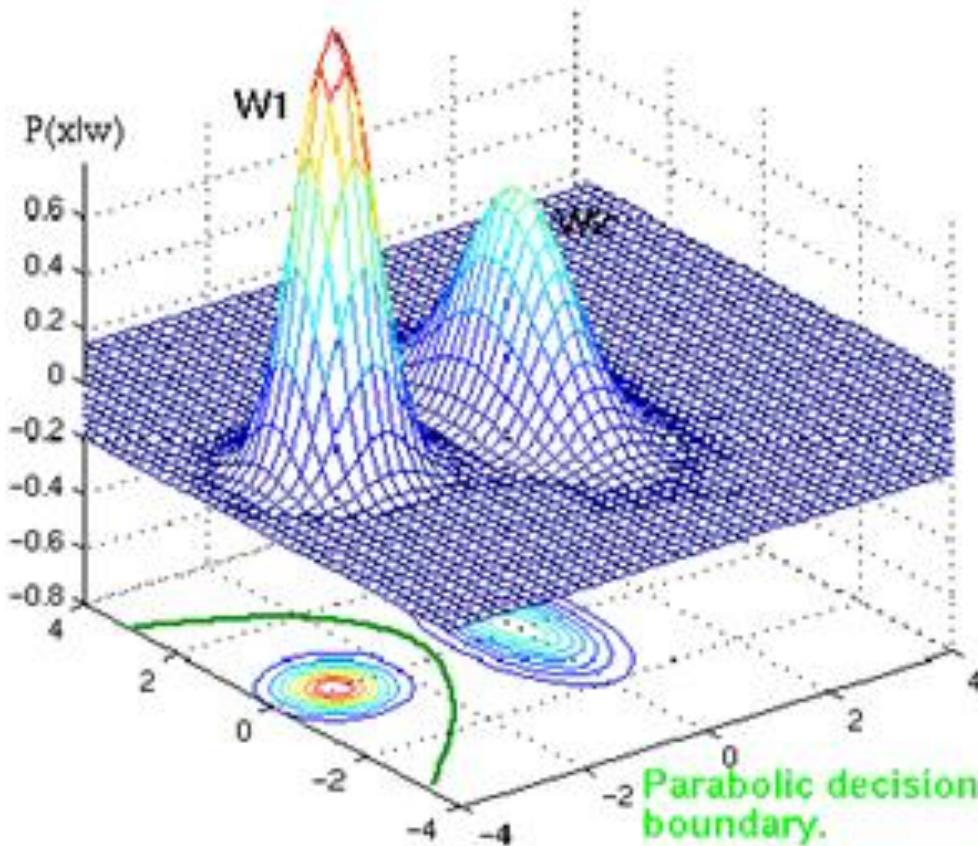
Нормальный дискриминантный анализ

Он же – квадратичный дискриминантный анализ (QDA)

$$a(x) = \operatorname{argmax}_y p(x|y)P(y)$$

$$p(x|y) = \frac{1}{(2\pi)^{\frac{n}{2}} \sqrt{\det \Sigma_y}} e^{-\frac{1}{2}(x-\mu_y)^T \Sigma_y^{-1} (x-\mu_y)}$$

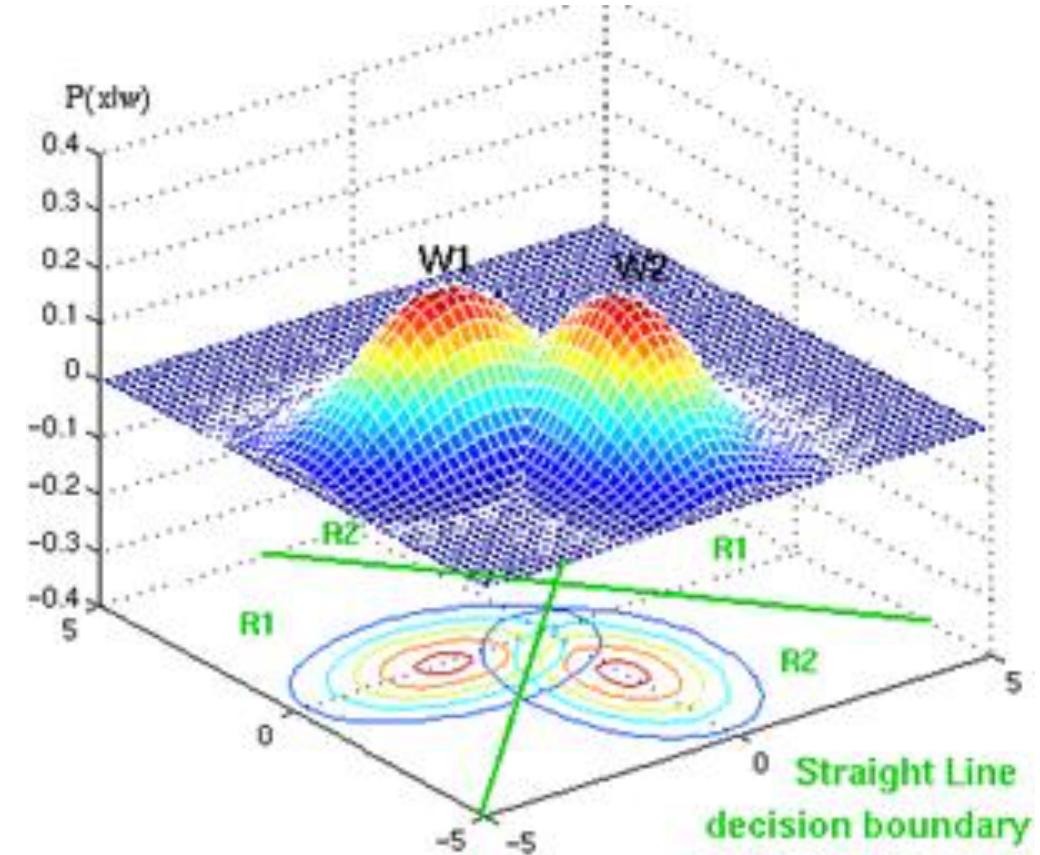
Нормальный дискриминантный анализ



Обычно разделяющая
поверхность имеет
параболическую,
гиперболическую или
эллиптическую форму

Вырождение в две прямые

В случае двух одинаковых по форме гауссиан, по-разному ориентированных в пространстве, разделяющая поверхность может принять вид двух прямых



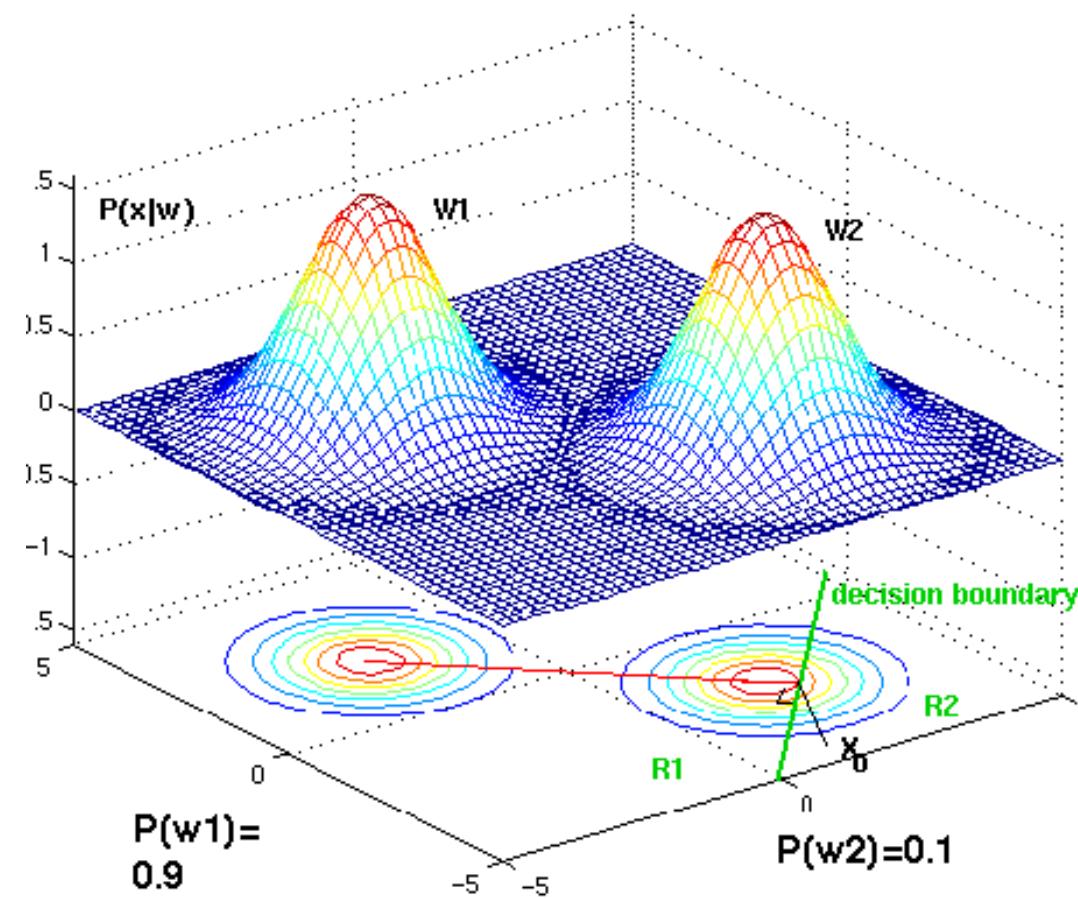
Линейный дискриминантный анализ

Он же – линейный дискриминант Фишера:

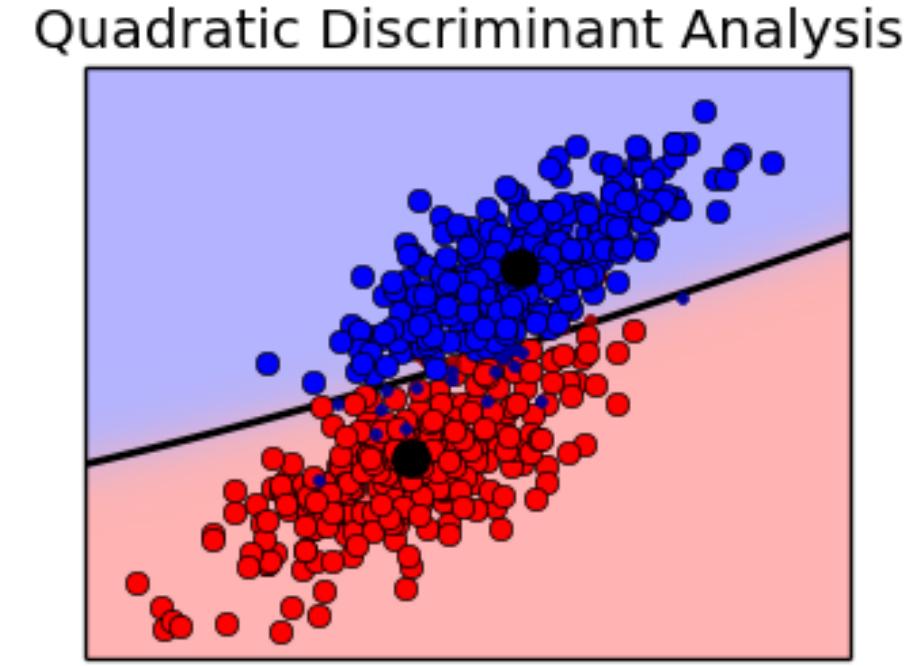
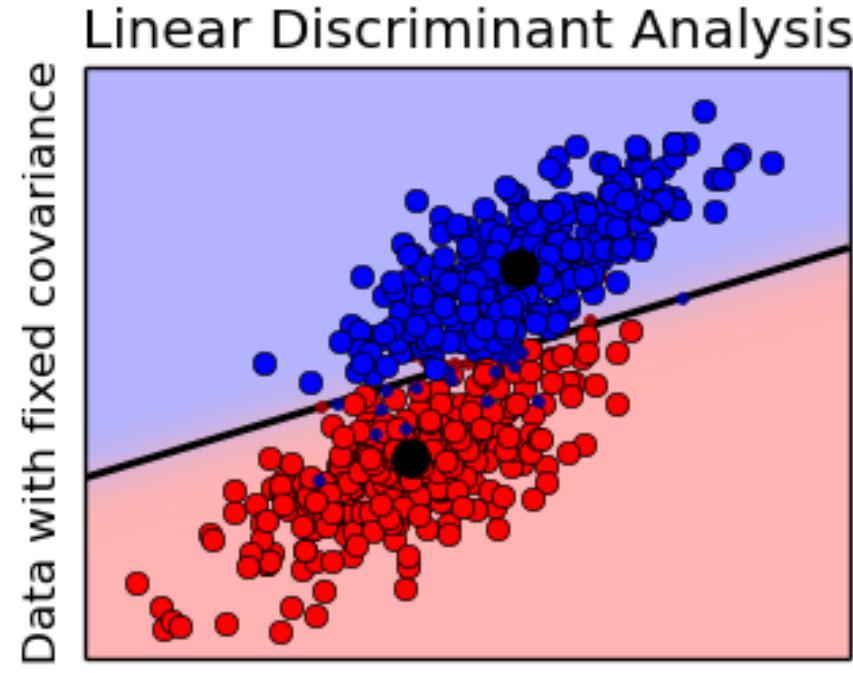
$$a(x) = \operatorname{argmax}_y p(x|y)P(y)$$

$$p(x|y) = \frac{1}{(2\pi)^{\frac{n}{2}} \sqrt{\det \Sigma}} e^{-\frac{1}{2}(x-\mu_y)^T \Sigma^{-1} (x-\mu_y)}$$

Линейный дискриминантный анализ

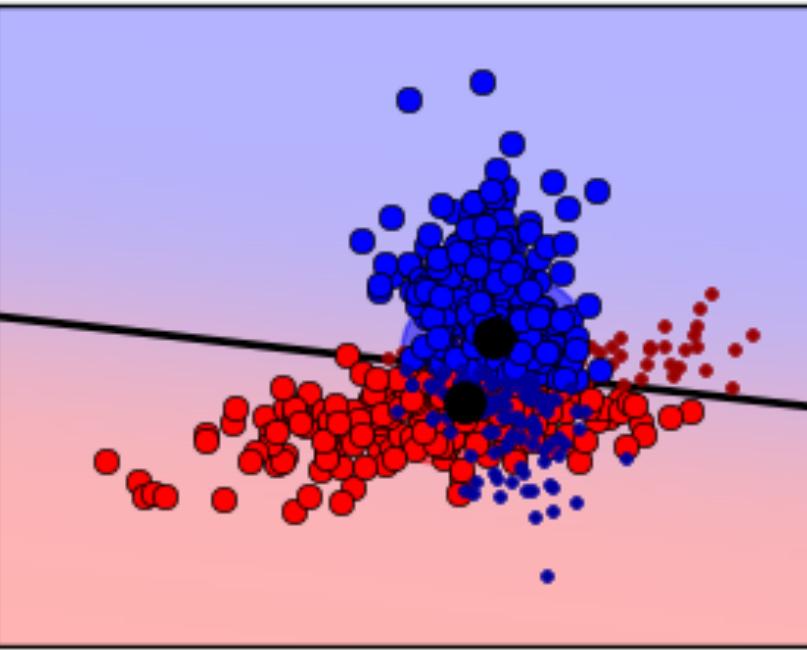


LDA vs QDA

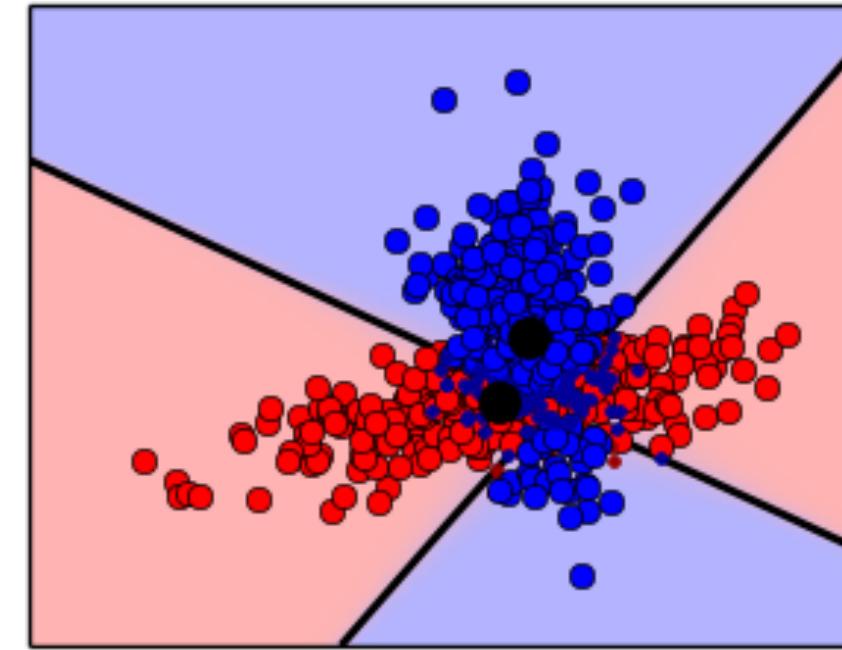


LDA vs QDA

Data with varying covariances



Quadratic Discriminant Analysis



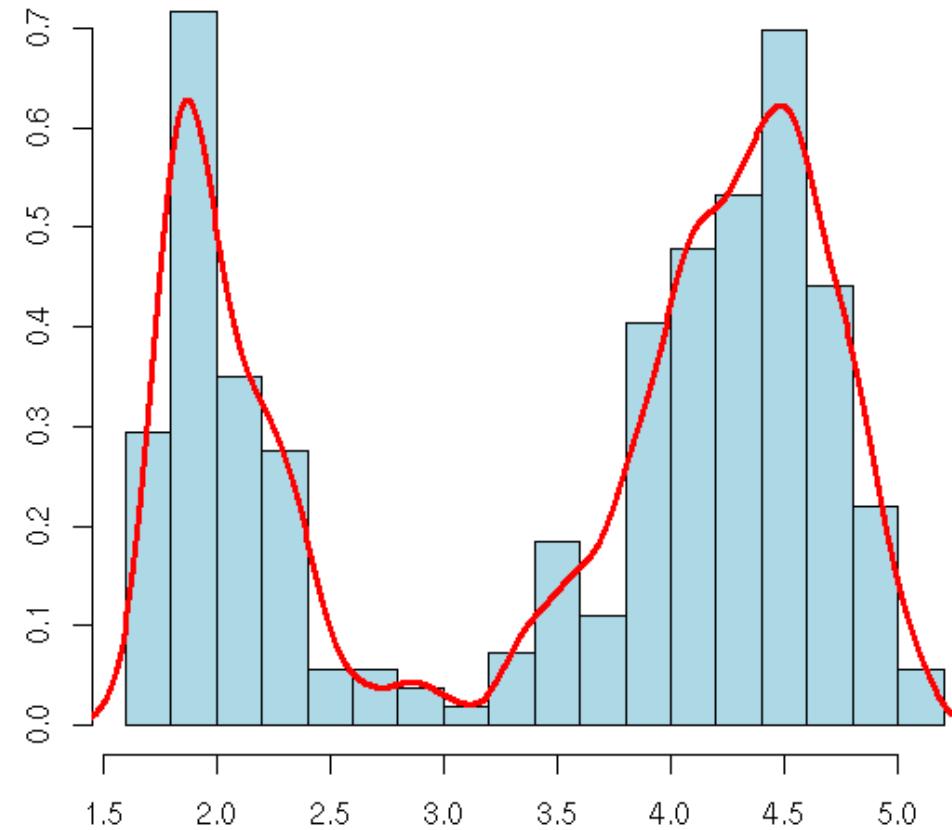
Недостатки подхода

- В QDA возникает больше параметров, чем в «наивном» подходе
- Для нормального распределения: p средних и p дисперсий в наивном подходе против вектора средних размерности p и матрицы ковариаций $p \times p$
- Оценка параметров может получиться неправильной из-за нехватки данных
- Часто требуется выполнять «неустойчивые» операции – например, обращение матриц, которые почти вырождены
- LDA – неплохой линейный классификатор, но обычно проигрывает логистической регрессии

Решение 3* – Непараметрическое оценивание

- Оценивать $P(x|y)$ можно не в точке, а в точке и ее окрестности
- Примеры, которые ближе к точке – учитывать с большим весом, те, которые дальше – с меньшим

Непараметрическое восстановление плотности



Оценка плотности: резюме

1. Проблема нехватки данных для восстановления распределений
2. Наивный байесовский классификатор
3. Параметрическая оценка многомерной плотности: QDA и LDA
4. Непараметрическая оценка многомерной плотности*

III. Связь функций потерь и распределения данных

Минимизация эмпирического риска

Оптимационная задача, решаемая при использовании функции потерь L :

$$Q = \sum_{i=1}^l L(y_i, a(x_i)) \rightarrow \min_w$$

Минимизация эмпирического риска

Оптимационная задача, решаемая при использовании функции потерь L :

$$-Q = \sum_{i=1}^l -L(y_i, a(x_i)) \rightarrow \max_w$$

Минимизация эмпирического риска

Оптимационная задача, решаемая при использовании функции потерь L :

$$-Q = \sum_{i=1}^l \ln e^{-L(y_i, a(x_i))} \rightarrow \max_w$$

Максимизация правдоподобия

Оптимационная задача, решаемая при использовании функции потерь L :

$$\prod_{i=1}^l e^{-L(y_i, a(x_i))} \rightarrow \max_w$$

$$p(y|x, a) \propto e^{-L(y_i, a(x_i))}$$

Максимизация правдоподобия

Оптимационная задача, решаемая при использовании функции потерь L :

$$\prod_{i=1}^l e^{-L(y_i, a(x_i))} \rightarrow \max_w$$

$$p(y|x, a) \propto e^{-L(y_i, a(x_i))}$$

Пример:

$$L(y_i, a(x_i)) = (y_i - a(x_i))^2, p(y|x, a) \propto e^{-(y_i - a(x_i))^2}$$

VI. Оптимизация риска и анализ функций потерь

Байесовский классификатор

$$a(x) = \operatorname{argmax}_y P(y)P(x|y)$$

x – признаковое описание

y - класс

Байесовская регрессия

$$a(x) = \operatorname{argmax}_y P(y)P(x|y) ?$$

x – признаковое описание

y – прогнозируемая величина

Байесовская регрессия

$$a(x) = \operatorname{argmax}_y P(y)P(x|y) ?$$

x – признаковое описание

y – прогнозируемая величина

Вряд ли получится восстановить $P(x|y)$

$a(x) = \operatorname{argmax}_y P(y|x)$ тоже сомнительный вариант для регрессии

Штрафы за ошибки

- В классификации
 - Разные ошибки классификации могут быть в разной степени критичны
 - Пример: классификация мест, в которых может быть обнаружено месторождение нефти на классы «есть нефть» и «нет нефти».
 - Можем назначить разные штрафы за разные ошибки

Штрафы за ошибки

- В классификации
 - Разные ошибки классификации могут быть в разной степени критичны
 - Пример: классификация мест, в которых может быть обнаружено месторождение нефти на классы «есть нефть» и «нет нефти».
 - Можем назначить разные штрафы за разные ошибки
- В регрессии
 - Зависимость в любом случае восстанавливается неточно
 - Квадратичные потери:

$$MSE = \frac{1}{l} \sum_{i=1}^l (y_i - a(x_i))^2$$

- Сумма модулей отклонений:

$$MAE = \frac{1}{l} \sum_{i=1}^l |y_i - a(x_i)|$$

Более общий подход

- Для объекта x мы делаем прогноз $a(x)$
- Правильный ответ на этом объекте y
- Величина ошибки алгоритма $L(y, a(x))$ (функцию выбираем сами)
- Пример 1: для классификации можно взять $L(y, a(x)) = [y \neq a(x)]$
- Пример 2: для регрессии подойдет $L(y, a(x)) = (y - a(x))^2$

Функционал риска

$$R(a(x), x) = \mathbb{E}(L(y, a(x)) \mid x)$$

Можно давать на объекте x ответ, который минимизирует ожидаемую ошибку:

$$a(x) = \operatorname{argmin}_s R(s, x)$$

Оптимальный байесовский классификатор

Для классификации:

$$R(a(x), x) = \mathbb{E}(L(y, a(x))|x) = \sum_{y \in Y} L(y, a(x))P(y|x)$$

$$\begin{aligned} a(x) &= \arg \min_s R(s, x) = \arg \min_s \sum_{y \in Y} L(y, s)P(y|x) = \\ &= \boxed{\arg \min_s \sum_{y \in Y} L(y, s)P(y)P(x|y)} \end{aligned}$$

Реальный классификатор в точности оптимальным не будет из-за погрешности в восстановлении плотностей

Оптимальный байесовский классификатор

$$a(x) = \operatorname{argmin}_s \sum_{y \in Y} L(s, y) P(y) P(x|y)$$

Частный случай

Если $L(s, y) = [y \neq s]$:

$$\sum_{y \in Y} L(s, y) P(y|x) \rightarrow \min$$

Частный случай

Если $L(s, y) = [y \neq s]$:

$$\sum_{y \in Y} L(s, y) P(y|x) \rightarrow \min$$

$$\sum_{y \in Y \setminus \{s\}} P(y|x) = \left(\sum_{y \in Y} P(y|x) \right) - P(s|x) \rightarrow \min$$

Частный случай

Если $L(s, y) = [y \neq s]$:

$$\sum_{y \in Y} L(s, y) P(y|x) \rightarrow \min_s$$

$$\sum_{y \in Y \setminus \{s\}} P(y|x) = \left(\sum_{y \in Y} P(y|x) \right) - P(s|x) \rightarrow \min$$
$$P(s|x) \rightarrow \max$$

Частный случай

$$a(x) = \arg \min_y P(y|x) = \arg \min_y P(y)P(x|y)$$

Оптимальный байесовский регрессор

Для регрессии:

$$R(a(x), x) = \mathbb{E}(L(y, a(x))|x) = \int_{y \in Y} L(y, a(x))p(y|x)dy$$

$$a(x) = \arg \min_s R(s, x) = \boxed{\arg \min_s \int_{y \in Y} L(y, s)p(y|x)dy}$$

Квадратичная функция потерь в регрессии

$$\int_Y (t - y)^2 p(y|x) dy \rightarrow \min_t$$

Квадратичная функция потерь в регрессии

$$\int_Y (t - y)^2 p(y|x) dy \rightarrow \min_t$$
$$\frac{\partial}{\partial t} \int_Y (t - y)^2 p(y|x) dy = 2 \int_Y (t - y) p(y|x) dy =$$

Квадратичная функция потерь в регрессии

$$\begin{aligned} & \int_Y (t - y)^2 p(y|x) dy \rightarrow \min_t \\ & \frac{\partial}{\partial t} \int_Y (t - y)^2 p(y|x) dy = 2 \int_Y (t - y) p(y|x) dy = \\ & = 2 \left(\int_Y t p(y|x) dy - \int_Y y p(y|x) dy \right) = 0 \end{aligned}$$

Квадратичная функция потерь в регрессии

$$\int_Y (t - y)^2 p(y|x) dy \rightarrow \min_t$$

$$\begin{aligned} \frac{\partial}{\partial t} \int_Y (t - y)^2 p(y|x) dy &= 2 \int_Y (t - y) p(y|x) dy = \\ &= 2 \left(\int_Y t p(y|x) dy - \int_Y y p(y|x) dy \right) = 0 \end{aligned}$$

$$a(x) = t = \int_Y y p(y|x) dy = \mathbb{E}(y|x)$$

Абсолютное отклонение

$$\int_Y |t - y| p(y|x) dy \rightarrow \min_t$$

Абсолютное отклонение

$$\int_Y |t - y| p(y|x) dy \rightarrow \min_t$$

$$\frac{\partial}{\partial t} \int_Y |t - y| p(y|x) dy = \frac{\partial}{\partial t} \int_{Y \setminus \{t\}} |t - y| p(y|x) dy =$$

Абсолютное отклонение

$$\int_Y |t - y| p(y|x) dy \rightarrow \min_t$$

$$\frac{\partial}{\partial t} \int_Y |t - y| p(y|x) dy = \frac{\partial}{\partial t} \int_{Y \setminus \{t\}} |t - y| p(y|x) dy =$$

$$= \int_{Y \setminus \{t\}} \text{sign}(t - y) p(y|x) dy = \int_{\{t > y\}} p(y|x) dy - \int_{\{t < y\}} p(y|x) dy =$$

Абсолютное отклонение

$$\begin{aligned} & \int_Y |t - y| p(y|x) dy \rightarrow \min_t \\ & \frac{\partial}{\partial t} \int_Y |t - y| p(y|x) dy = \frac{\partial}{\partial t} \int_{Y \setminus \{t\}} |t - y| p(y|x) dy = \\ & = \int_{Y \setminus \{t\}} \text{sign}(t - y) p(y|x) dy = \int_{\{t > y\}} p(y|x) dy - \int_{\{t < y\}} p(y|x) dy = \\ & = \mathsf{P}(\{t > y\}|x) - \mathsf{P}(\{t < y\}|x) = 0. \end{aligned}$$

Абсолютное отклонение

$$\int_Y |t - y| p(y|x) dy \rightarrow \min_t$$

$$\frac{\partial}{\partial t} \int_Y |t - y| p(y|x) dy = \frac{\partial}{\partial t} \int_{Y \setminus \{t\}} |t - y| p(y|x) dy =$$

$$= \int_{Y \setminus \{t\}} \text{sign}(t - y) p(y|x) dy = \int_{\{t > y\}} p(y|x) dy - \int_{\{t < y\}} p(y|x) dy = \\ = \mathsf{P}(\{t > y\}|x) - \mathsf{P}(\{t < y\}|x) = 0.$$

$$\mathsf{P}(\{t = y\}|x) = 0$$

Абсолютное отклонение

$$\begin{aligned} & \int_Y |t - y| p(y|x) dy \rightarrow \min_t \\ & \frac{\partial}{\partial t} \int_Y |t - y| p(y|x) dy = \frac{\partial}{\partial t} \int_{Y \setminus \{t\}} |t - y| p(y|x) dy = \\ & = \int_{Y \setminus \{t\}} \text{sign}(t - y) p(y|x) dy = \int_{\{t > y\}} p(y|x) dy - \int_{\{t < y\}} p(y|x) dy = \\ & = \mathsf{P}(\{t > y\}|x) - \mathsf{P}(\{t < y\}|x) = 0. \end{aligned}$$

$$\mathsf{P}(\{t = y\}|x) = 0 \implies P(\{t \leq y\}|x) = P(\{t > y\}|x) = \frac{1}{2}$$

Оценка вероятностей

$$Y = \{0; 1\} \quad L(y, a(x)) = -y \ln a(x) - (1 - y) \ln(1 - a(x))$$

Оценка вероятностей

$$Y = \{0; 1\} \quad L(y, a(x)) = -y \ln a(x) - (1 - y) \ln(1 - a(x))$$
$$\sum_{y \in Y} (-y \ln t - (1 - y) \ln(1 - t)) P(y|x) \rightarrow \min_t$$

Оценка вероятностей

$$Y = \{0; 1\} \quad L(y, a(x)) = -y \ln a(x) - (1 - y) \ln(1 - a(x))$$

$$\sum_{y \in Y} (-y \ln t - (1 - y) \ln(1 - t)) P(y|x) \rightarrow \min_t$$

$$\mathsf{P}(1|x) = p$$

Оценка вероятностей

$$Y = \{0; 1\} \quad L(y, a(x)) = -y \ln a(x) - (1 - y) \ln(1 - a(x))$$

$$\sum_{y \in Y} (-y \ln t - (1 - y) \ln(1 - t)) P(y|x) \rightarrow \min_t$$

$$P(1|x) = p \quad -(1 - p) \ln(1 - t) - p \ln t \rightarrow \min_t$$

Оценка вероятностей

$$Y = \{0; 1\} \quad L(y, a(x)) = -y \ln a(x) - (1 - y) \ln(1 - a(x))$$

$$\sum_{y \in Y} (-y \ln t - (1 - y) \ln(1 - t)) P(y|x) \rightarrow \min_t$$

$$P(1|x) = p \quad -(1 - p) \ln(1 - t) - p \ln t \rightarrow \min_t$$

$$\frac{\partial}{\partial t}(-(1 - p) \ln(1 - t) - p \ln t) = \frac{1 - p}{1 - t} - \frac{p}{t} =$$

Оценка вероятностей

$$Y = \{0; 1\} \quad L(y, a(x)) = -y \ln a(x) - (1 - y) \ln(1 - a(x))$$

$$\sum_{y \in Y} (-y \ln t - (1 - y) \ln(1 - t)) P(y|x) \rightarrow \min_t$$

$$P(1|x) = p \quad -(1 - p) \ln(1 - t) - p \ln t \rightarrow \min_t$$

$$\frac{\partial}{\partial t}(-(1 - p) \ln(1 - t) - p \ln t) = \frac{1 - p}{1 - t} - \frac{p}{t} =$$

$$= \frac{(1 - p)t - p(1 - t)}{(1 - t)t} = \frac{t - p}{(1 - t)t} = 0$$

Оценка вероятностей

$$Y = \{0; 1\} \quad L(y, a(x)) = -y \ln a(x) - (1 - y) \ln(1 - a(x))$$

$$\sum_{y \in Y} (-y \ln t - (1 - y) \ln(1 - t)) P(y|x) \rightarrow \min_t$$

$$P(1|x) = p \quad -(1 - p) \ln(1 - t) - p \ln t \rightarrow \min_t$$

$$\frac{\partial}{\partial t}(-(1 - p) \ln(1 - t) - p \ln t) = \frac{1 - p}{1 - t} - \frac{p}{t} =$$

$$= \frac{(1 - p)t - p(1 - t)}{(1 - t)t} = \frac{t - p}{(1 - t)t} = 0 \Rightarrow t = p$$

Почему это все работает

- Средний риск:

$$R(a) = \mathbb{E}_{x,y} L(y, a(x))$$

Почему это все работает

- Средний риск:

$$R(a) = \mathbb{E}_{x,y} L(y, a(x))$$

- Ошибка на обучающей выборке:

$$Q = \frac{1}{l} \sum_{i=1}^l L(y_i, a(x_i)) \approx \mathbb{E}_{x,y} L(y, a(x))$$

Средний риск и риск

- Средний риск:

$$R(a) = \mathbb{E}_{x,y} L(y, a(x))$$

- Риск на объекте x :

$$R(a, x) = \mathbb{E}_{y|x} L(y, a(x))$$

Утверждение: если на каждом объекте отвечать, минимизируя риск, полученный алгоритм будет минимизировать функционал среднего риска

Резюме

- Принцип минимизации функционала риска
- Анализ функций потерь
- Квадратичная функция потерь – для оценки матожидания
- Абсолютные отклонения – для оценки $\frac{1}{2}$ квантили
- Log loss – для оценки вероятностей
- Понимание неудачного выбора функции потерь

Спасибо за внимание



info@applieddatascience.ru



https://t.me/joinchat/B10lThC96v0BQCvs_joNew



https://github.com/vkantor/ml2018jan_feb



<https://goo.gl/forms/TRyruBwJplBvnfzG3>

Функционал среднего риска

- Можно рассмотреть $R(a) = \mathbb{E}_x R(a(x), x)$ (по всем x из X)
- Для определенности рассмотрим случай классификации объектов с дискретными признаками:

$$R(a) = \sum_{x \in X} R(a(x), x) P(x)$$

- Можно оценить $R(a)$ снизу:

$$\sum_{x \in X} R(a(x), x) P(x) \geq \sum_{x \in X} P(x) \min_s R(s, x)$$

Функционал среднего риска

1. Можно рассмотреть $R(a) = \mathbb{E}_x R(a(x), x)$ (по всем x из X)
2. Для определенности рассмотрим случай классификации объектов с дискретными признаками:

$$R(a) = \sum_{x \in X} R(a(x), x) P(x)$$

3. Можно оценить $R(a)$ снизу:

$$\sum_{x \in X} R(a(x), x) P(x) \geq \sum_{x \in X} P(x) \min_s R(s, x)$$

Если $a(x)$ – оптимальный байесовский, он минимизирует $R(a(x), x)$
Значит оценка достигается и $R(a)$ он тоже минимизирует