

kNN, деревья решений, предобработка

Маша Шеянова, masha.shejanova@gmail.com

Формула оцѣнки

Оцѣнка за курс = $0.5 * \text{накоп} + 0.5 * \text{проект}$

Накоп = 2 маленьких дз * 0.2 + 2 больших дз * 0.8 + конспект статьи на медиуме (по желанию)

Популярные алгоритмы классификации

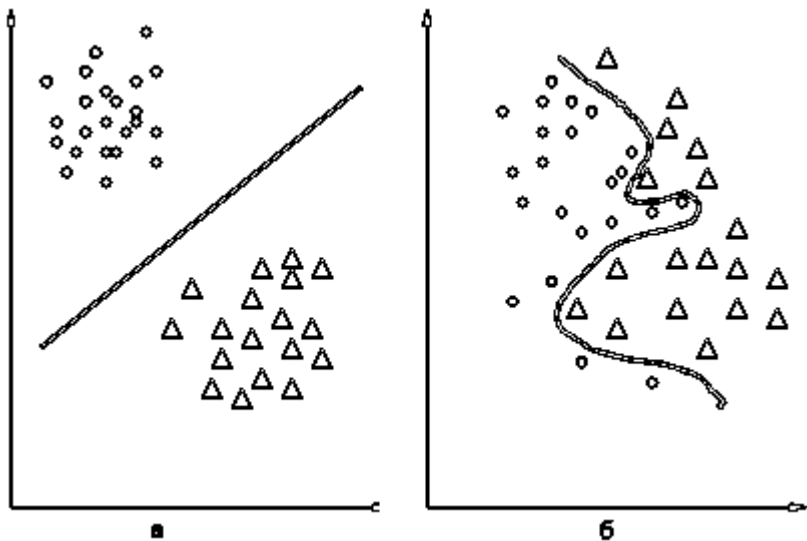
- **к ближайших соседей (kNN)**
- наивный Байес
- **деревья решений**
- логистическая регрессия
- метод опорных векторов (SVM)

На прошлой лекции мы обсуждали наивный байесовский классификатор. Теперь разберёмся в kNN и деревьях решений.

Метрические классификаторы. kNN.

Гипотеза компактности

Это предположение о том, что схожие (близкие в пространстве признаков) объекты гораздо чаще лежат в одном классе, чем в разных.



Метрики (математика)

(не путать с метриками качества, они никак не связаны!)

Метрики — это такой способ посчитать расстояние между объектами. Иными словами, ‘это функция $d(a, b)$, значение которой — расстояние от a до b .

Для метрик выполняются такие правила:

- $d(a, b) = 0$, iff $a = b$ — аксиома тождества
- $d(a, b) = d(b, a)$ — аксиома симметрии
- $d(a, b) \leq d(a, c) + d(b, c)$ — неравенство треугольника

Функция расстояния

Для того, чтобы сделать метрический классификатор, надо уметь считать расстояние между объектам; находить, какой ближе, а какой дальше.

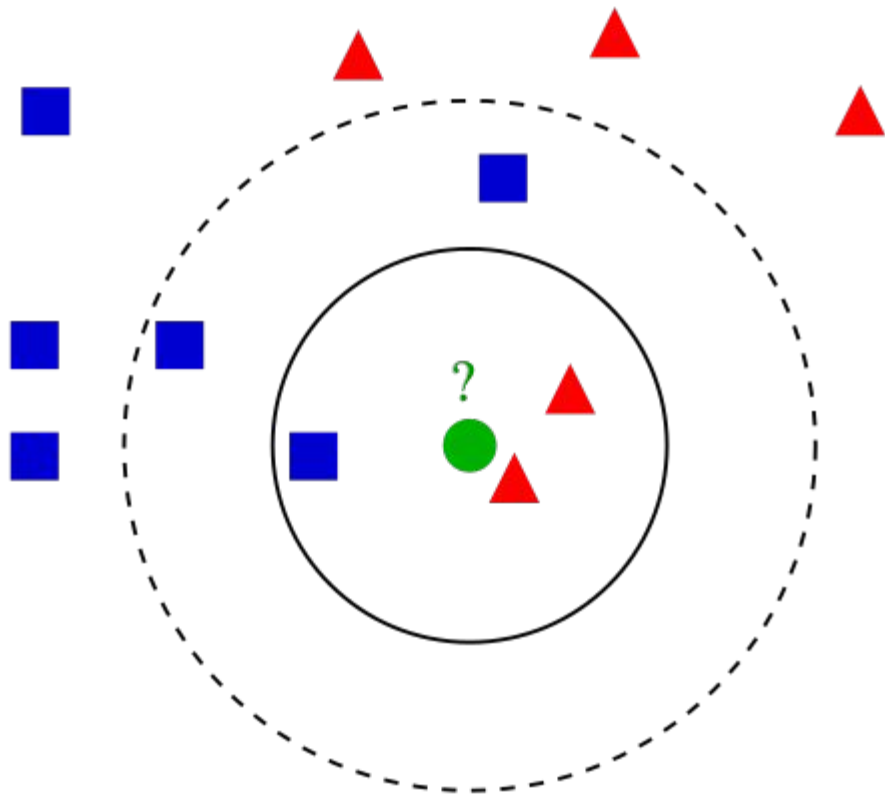
Метрики (в математическом смысле) — хорошие функции расстояния.

Но как выглядят наши объекты? Это вектора! Каждая координата — значение того или иного признака.

А значит, в качестве функции расстояния можно воспользоваться:

- евклидовым расстоянием
- косинусным расстоянием

k ближайших соседей (kNN)



Источник картинки.

К какому классу будет отнесён
кружок в центре при $k=3$?

При $k=5$?

kNN: преимущества и недостатки

Преимущества:

- соседи “голосуют” — вывод о классе чувствителен к шуму
- параметр k можно настраивать для каждой задачи

Недостатки:

- непонятно, что делать если среди k соседей одинаковое количество представителей разных классов

kNN в sklearn

[sklearn.neighbors.KNeighborsClassifier](#)

Можно настраивать параметры:

- `n_neighbors` (по дефолту, 5)
- `weights` (по дефолту, “uniform” — ; ещё может быть “distance” и функция)
- `algorithm` — разные реализации

И ещё несколько других параметров.

Деревья решений

Идея

Каждый признак — критерий, чтобы выбрать, к какому классу относится объект. Мы можем построить **дерево**, где каждый **узел — разветвление по признаку**. Корень — самый значимый признак, дальше другие признаки.

Плюсы:

- очень интуитивно

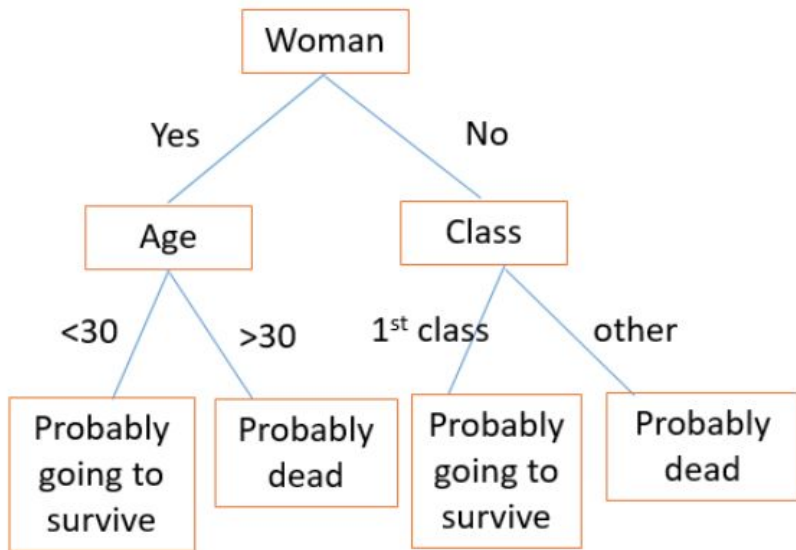
Минусы:

- склонны к переобучению

Дерево решений

Источник картинки.

Дерево решений на примере датасета из титанка.



Энтропия

Один из алгоритмов — ID3 (Iterative Dichotomiser 3), использует энтропию.


Как измерить насколько распределение "неопределённое"?

Взять математическое ожидание количества бит, которое понадобится, чтобы закодировать один из исходов.

Это количество бит — **информация**
($-\log p$).

Мат ожидание информации - **энтропия**.

$$Entropy = - \sum p(X) \log p(X)$$



here $p(x)$ is a fraction of
examples in a given class

Information gain

*“**Information gain (IG)** measures how much “information” a feature gives us about the class.”*

Сколько информации вносит родительский узел:

$$\text{Information gain} = \text{entropy (parent)} - [\text{weightes average}] * \text{entropy (children)}$$

Gini

... to be explained

Деревья решений в sklearn

[sklearn.tree.DecisionTreeClassifier](#)

Параметры:

- criterion (gini или entropy)
- splitter (best или random)
- max_depth — максимально возможная высота дерева
- max_leaf_nodes — максимально возможная “ширина” дерева

И другие.

Ещё немного про метрики

Точность и полнота для нескольких классов

Мы обсудили, что точность и полнота строится по такой табличке:

Но что если классов много — получается, они не работают?

А вот и работают: надо для каждого класса считать,

Что такое baseline

Это тот результат, с которым вы сравниваете свой метод. Лучший результат, который можно было получить простым способом.

Например, в задаче бинарной классификации со сбалансированными классами можно “подбрасывать монетку”: accuracy будет ~ 0.5 .

Если вы изобрели новый метод для решения задачи, baseline — прошлый лучший метод. Если вы сравниваете, насколько важна лемматизация, baseline — результат без лемматизации.

Задача: придумайте простой baseline для определения оскорбительных твитов, для которого не нужно МО.

О предобработке

Что можно сделать с текстом?

Предобработка — в принципе, все изменения, которые вы делаете с данными до того, как извлечь из них признаки.

- почистить текст от мусора (например, от остатков markdown)
- убрать стоп-слова (*а, не, на, и, ...*), пунктуацию
- сделать умную токенизацию
- лемматизировать слова
- добавить информацию о частях речи
- добавить информацию о роли в предложении
- ...

Ресурсы

Почитать

- [про деревья решений](#) (англ)
- [про энтропию и information gain для деревьев решений](#) (англ)
- [про kNN](#) (англ)