Базовые модели машинного обучения: k-ближайших соседей

Гирдюк Дмитрий Викторович
Першин Антон Юрьевич, Ph.D.
Никольская Анастасия Николаевна

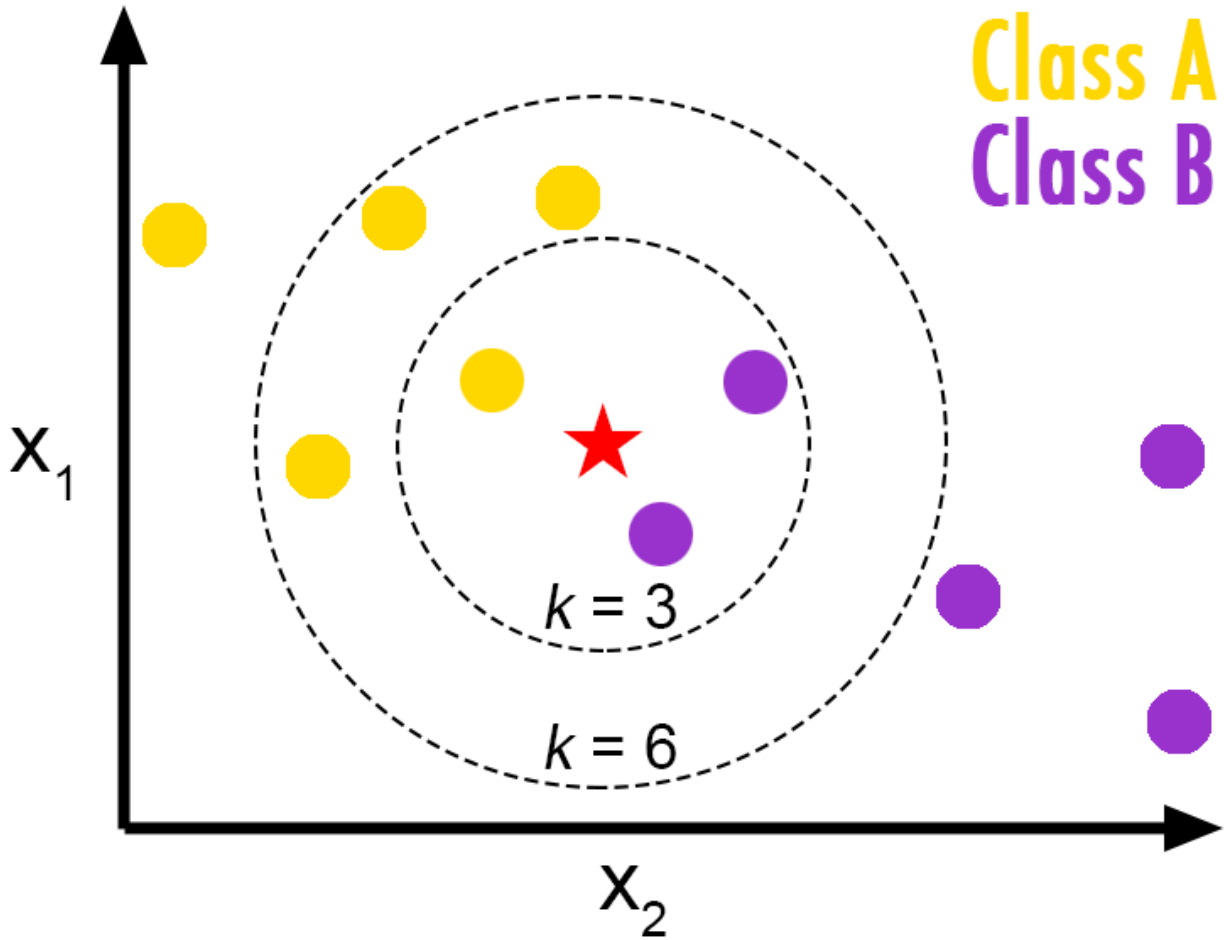
Программа «Большие данные и распределенная цифровая платформа»
Санкт-Петербургский государственный университет

Практика по дисциплине «Технологии ИИ»
15 апреля 2023 г.

Метод k-ближайших соседей

- Метод k-ближайших соседей (k-nearest neighbors, kNN) – относительно простой метрический алгоритм для задач классификации, основанный на оценке схожести некоторого наблюдения/объекта/сэмпла и классифицированных ранее его соседей.
- Классифицируемое наблюдение относится к классу, преобладающему среди k ближайших соседей наблюдения.
- Близость определяется некоторой фиксированной метрикой (например, евлкидовой).
- Основное предположение заключается в том, что близкие наблюдения (в смысле значения метрики) принадлежат одному классу (так называемая «гипотеза компактности»).

Пример



Источник изображения

Формальное определение

→ Имеем размеченную обучающую выборку $X = \{x_i\}_{i=1}^n$, $Y = \{y_i\}$, $x_i \in \mathbb{R}^m$, $y_i \in \mathbb{N}$.

→ Выберем некоторую метрику $\rho(x_i, x_j)$.

→ Отсортируем для некоторого нового наблюдения $\widehat{x}$ объекты обучающей выборки X:

$$\rho(\widehat{x}, x_{o_1}) \leqslant \rho(\widehat{x}, x_{o_2}) \leqslant \ldots \leqslant \rho(\widehat{x}, x_{o_n})$$

→ Тогда метод ближайших соседей формально записывается в виде

$$\widehat{y} = \arg\max_{y \in Y} \sum_{i=1}^{n} [y_{o_i} = y]\, \omega(i, \widehat{x}),$$

где $\omega(i, \widehat{x})$ есть весовая функция, которая оценивает степень важности o_i-го наблюдения для классификации $\widehat{x}$.

→ $\omega(i, \widehat{x}) = [i = 1]$ – метод ближайшего соседа.

→ $\omega(i, \widehat{x}) = [i \leqslant k]$ – метод k-ближайших соседей.

Выбор k

$\rightarrow$ Очевидно, что при $k = 1$ метод является неустойчивым к выбросам, а при $k = n$ все новые наблюдения будут относиться к наиболее частотному классу.

$\rightarrow$ На практике k выбирается либо на основе внешних свойств исследуемой области, либо путем кросс-валидации.

Типы наблюдений

→ Наблюдения можно разделить на 3 типа.

→ Эталоны – самые информативные наблюдения, типичные представители своего класса.

→ Когда в некоторой области признакового пространства содержится большое количество эталонных наблюдений, многие из них становятся неинформативными: их удаление никоим образом не скажется на качестве классификации.

→ Наконец, выбросы. Под ними понимаются как наблюдения, достаточно далеко удаленные ото всех остальных, так и те, что находятся в пределах большого числа наблюдений другого класса.

→ Понятно, что чем меньше в обучающей выборке неинформативных наблюдений и выбросов, тем лучше качество классификации.

Масштабируемость

→ Чем больше обучающая выборка, тем дольше происходит классификация.

→ Если в решаемой задаче необходимо последовательное дообучение, вычисление расстояния до всех наблюдений становится весьма неэффективным.

→ В таком случае, необходимы эффективные реализации поиска соседей на основе специфических структур данных/индексов (например, KD-деревья), или вовсе специальные схемы аппроксимации (например, Hierarchical Navigable Small Worlds, HNSW).

Выбор метрики ρ

→ Метрика должна достаточно адекватно отражать схожесть наблюдений в признаковом пространстве. Проблема состоит в том, что понятие “адекватно” сложно формализовать.

→ Числовые признаки практически всегда необходимо нормализовывать. Иначе вклад одних будет затмевать другие. С другой стороны, некоторые признаки могут быть куда более значимыми, чем другие.

→ Проклятие размерности тоже никто не отменял. Если признаков много, то сумма отклонений между компонентами двух наблюдений приведет к тому, что большинство наблюдений будут равноудалены относительно друг друга (см. закон больших чисел). Зато можно брать произвольное k!

→ Отсюда следует, что либо признаки следует каким-либо образом отбирать, либо задавать им в метрике весовые коэффициенты. Или вовсе «обучать метрику» (см. metric learning) под признаковое пространство.

Обсуждение

→ Метод k-ближайших соседей, наряду с деревьями, отличное базовое решение.

→ kNN имеет всего 2 гиперпараметра, каждый из которых имеет принципиальное значение.

→ Как и деревья решений, обобщается на задачи регрессии: значение вычисляется как среднее значений по соседям.

kNN в scikit-learn

- → kNN реализован в scikit-learn: KNeighborsClassifier и KNeighborsRegressor.
- → Есть поддержка разреженных данных.
- → Кроме числа соседей k можно задавать следующее:
  - weights, веса наблюдений. Либо равнозначны (дефолтное), либо с учетом расстояния до соседей.
  - algorithm. Способ поиска соседей: брутфорс, ball-дерево, KD-дерево, и автоматический подбор подходящего с учетом обучающей выборки (дефолтное).
  - leaf_size. Максимальный размер листа в дереве, если выбрано ball/KD-дерево.
  - metric и p. Метрику можно как реализовать самостоятельно, так и использовать одну из имеющихся: расстояние Минковского (p – ее параметр) и его частные случаи (Чебышева и Манхэттенская).