Метод к-го ближайшего соседа (K-Nearest Neighbors, kNN)

Прекрасные математические свойства.

Доминировал ранее при создании рекомендательных сервисов.

Сейчас метод вытеснен другими, почти не применяется.

Но еще не вечер.

Предложен в работе

Fix, E., Hodges, J.L. (1951)

Discriminatory analysis, nonparametric discrimination: Consistency properties.

Technical Report 4, USAF School of Aviation Medicine, Randolph Field, Texas.

Метод k-го ближайшего соседа состоятельный

- Байесовский классификатор имеет максимальную точность (accuracy)
- Но он требует знания совместного распределения предикторов и отклика.
- Метод состоятельный, если его точность стремится к точности байесовского классификатора, когда число наблюдений стремится к бесконечности.

- Байесовский классификатор не имеет отношения ни к теореме Байеса, ни к байесовской статистике, ни к преподобному Томасу Байесу. Так вышло...

- На практике плохие результаты: негативное влияние малоинформативных переменных.

- Вытеснен другими методами: XGBoost, Deep Learning

- Все может измениться
- KNNImputer — заполнение пропусков

- ***Еврипид:***

  Скажи мне, кто твой друг, и я скажу кто ты

- ***Утиный тест (Дж У Райли):***

  Когда я вижу птицу, которая ходит как утка, плавает как утка и крякает как утка, я называю эту птицу уткой.

Метод k-го ближайшего соседа

k=1 (один сосед)

Объект относится к тому же классу, что и его ближайший сосед.

k>1

Объект относится к тому классу, члены которого составляют большинство среди *k* ближайших соседей.

Технические детали

Если невозможно найти ровно *k* соседей, значение *k* увеличивают (вариант: выбрать случайно).

Если невозможно определить класс, который составляет большинство среди *k* соседей, значение *k* увеличивают(вариант: выбрать случайно).

Если классов два, удобно использовать нечетные значения *k*.

Правильно увеличивать *k* с ростом объема обучающей выборки n.

Из общих соображений ясно, что k должно быть разным в зависимости от локальной плотности точек, но пока неизвестно каким...

Вероятность принадлежать классу

- Модификация knn метода, чтобы оценивать вероятность принадлежать классу
- Задача: если взвешивать соседей расстоянием, как получать вероятности?
- Нужна калибровка

Вероятность принадлежать классу. Пример

- K=9
- Распознаем 3 класса. Из 9 соседей
- Первые 5 принадлежат классу «1»,
- следующие 3 принадлежат классу «2»,
- последний принадлежит классу «3»
- Как определить вероятность принадлежать классу?
- Модифицировать определение, если соседям придаем веса, обратно пропорциональные расстояниям до них. Расстояния равны 5, 9, 6, 1, 3, 8, 4, 2, 3

модификация метода

каждому соседу приписывают вес, обратно пропорциональный расстоянию до него

Всего 2 параметра

Необходимо настраивать всего два параметра:

k - число ближайших соседей.

способ вычисления расстояния между объектами. Расстояние измеряет сходство объектов.

Никаких предположений о распределении данных (при доказательстве состоятельности предполагаем существование плотности)

При этом метод состоятельный!

lazy learning

Нет модели

Но храним в памяти всю обучающую выборку.

Обучение алгоритма сводится к хранению обучающей выборки, организованное "правильным" образом, например в виде kd-tree.

Новые наблюдения

легко обновлять после добавления новых наблюдений в обучающую выборку

Недостатки алгоритма

1. Большой объем вычислений.

Лекарство: хранить данные в виде R-tree или kd-tree.

(когда число наблюдений значительно превышает число переменных)

Недостатки алгоритма

2. Чувствительность к шуму и малоинформативным переменным

Лекарство: стандартизация и/или feature selection.

Важно!

Стандартизация переменных!

Отступление:

Feature selection

Пример 1 квадрат на 4 части

Пример 2 круг в кольце

Недостатки алгоритма

3. Чувствительность к несбалансированности объемов классов: доминирующий класс может доминировать и в большинстве окрестностей.

Лекарство: выравнивание количества наблюдений в разных классах на этапе обучения.

Как выбрать значение *k*

(или протестировать применимость feature selection)

n-fold cross-validation

Отступление про кросс-валидацию

KNN в задачах регрессии

Находим k ближайших соседей точки X .

Y = f(x) определяем как (взвешенное) среднее Y-ков у ближайших соседей.

Если нет координат точек, но

Если можем сосчитать расстояния между точками,

то можем распознавать методом KNN.

KNN в Python

- В scikit-learn 0.22.2
- neighbors.KneighborsClassifier
- neighbors.KneighborsRegressor
- sklearn.impute.KNNImputer

KNN в R

5+ пакетов реализуют метод kNN:

Knn $O(n)$

FNN $O(\log(n))$

knntree

knnflex

kknn