Математические и инструментальные методы машинного обучения

7. Мягкая и жесткая кластеризация

Мягкая кластеризация

Мягкая кластеризация (англ. *fuzzy clustering* и *soft clustering*) — тип кластеризации, при котором каждая точка может принадлежать одному или нескольким кластерам. Мягкая кластеризация также называется нечёткой кластеризацией и используется при решении задач обработки естественного языка, в том числе в лексической семантике.

DBSCAN – Density-Based Spatial Clustering

Основанная на плотности пространственная кластеризация для приложений с шумами (англ. *Density-based spatial clustering of applications with noise*, **DBSCAN**) — это алгоритм кластеризации данных, который предложили Маритин Эстер, Ганс-Петер Кригель, Ёрг Сандер и Сяовэй Су в 1996[1]. Это алгоритм кластеризации, основанной на плотности — если дан набор точек в некотором пространстве, алгоритм группирует вместе точки, которые тесно расположены (точки со многими близкими соседями[en]), помечая как выбросы точки, которые находятся одиноко в областях с малой плотностью (ближайшие соседи которых лежат далеко).

DBSCAN – Density-Based Spatial Clustering

Алгоритм DBSCAN может быть разложен на следующие шаги:

1. Находим точки в $epsilon$ окрестности каждой точки и выделяем основные точки с более чем $minPts$ соседями.
2. Находим связные компоненты основных точек на графе соседей, игнорируя все неосновные точки.
3. Назначаем каждую неосновную ближайшему кластеру, если кластер является $epsilon$ — соседним, в противном случае считаем точку шумом.
4. Наивная реализация алгоритма требует запоминания соседей на шаге 1, так что требует существенной памяти. Оригинальный алгоритм DBSCAN не требует этого за счёт того, что выполняет эти шаги для одной точки за раз.

DBSCAN – Density-Based Spatial Clustering

DBSCAN посещает каждую точку базы данных, может быть несколько раз (например, как кандидаты в другие кластеры). По опыту эксплуатации, однако, временна́я сложность в основном регулируются числом запросов $regionQuery$. DBSCAN выполняет в точности один такой запрос для каждой точки и, если используется индексная структура, которая выполняет запрос соседства[en] за время $\mathrm{O}(\log n)$, получаем полную среднюю временну́ю сложность $\mathrm{O}(n \log n)$ (если параметр epsilon выбирается осмысленно, то есть так, что в среднем возвращается только $\mathrm{O}(\log n)$ точек). Без использования ускоряющей индексной структуры или на вырожденных данных (например, когда все точки находятся на расстоянии меньше чем $epsilon$), худшим случаем времени работы остаётся $\mathrm{O}(n^2)$. Матрица расстояний размера $(n^2\text{-}n)/2$ может быть вычислена во избежание перевычисления расстояний, но это требует памяти $\mathrm{O}(n^2)\mathrm{O}(n^2)$, в то время как реализация DBSCAN без матрицы расстояний требуется лишь $\mathrm{O}(n)$ памяти.

DBSCAN – Density-Based Spatial Clustering

Преимущества

- DBSCAN не требует спецификации числа кластеров в данных априори в отличие от метода k-средних.
- DBSCAN может найти кластеры произвольной формы. Он может найти даже кластеры полностью окружённые (но не связанные с) другими кластерами. Благодаря параметру MinPts уменьшается так называемый эффект одной связи (связь различных кластеров тонкой линией точек).
- DBSCAN имеет понятие шума и устойчив к выбросам.
- DBSCAN требует лишь двух параметров и большей частью нечувствителен к порядку точек в базе данных. (Однако, точки, находящиеся на границе двух различных кластеров могут оказаться в другом кластере, если изменить порядок точек, а назначение кластеров единственно с точностью до изоморфизма.)

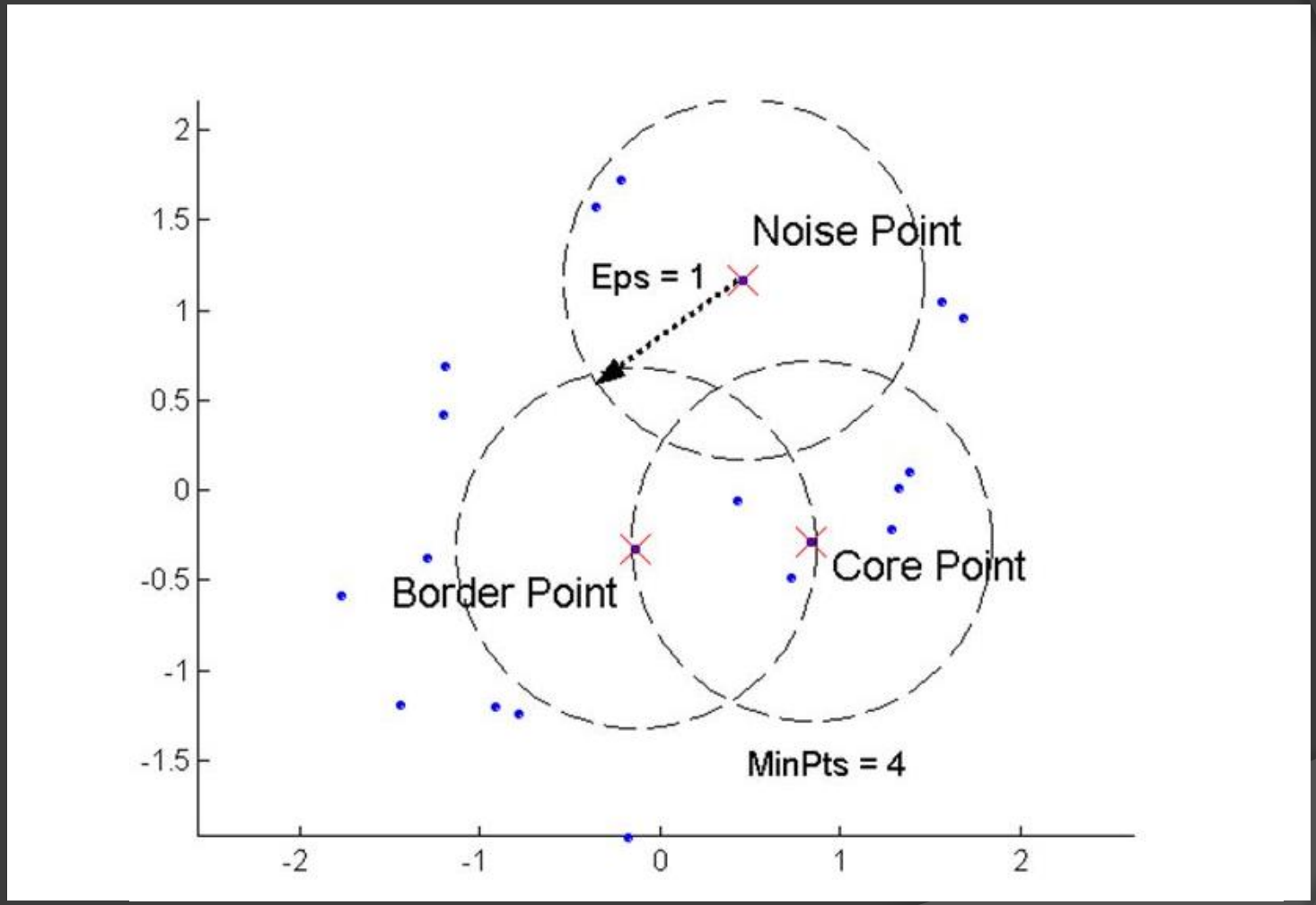
DBSCAN – Density-Based Spatial Clustering

- DBSCAN разработан для применения с базами данных, которые позволяют ускорить запросы в диапазоне значений, например, с помощью R^*-дерева.
- Параметры $minPts$ и epsilon могут быть установлены экспертами в рассматриваемой области, если данные хорошо понимаются.
- DBSCAN не полностью однозначен — краевые точки, которые могут быть достигнуты из более чем одного кластера, могут принадлежать любому из этих кластеров, что зависит от порядка просмотра точек. Для большинства наборов данных эти ситуации возникают редко и имеют малое влияние на результат кластеризации[6] — основные точки и шум DBSCAN обрабатывает однозначно. DBSCAN*[8] является вариантом, который трактует краевые точки как шум и тем самым достигается полностью однозначный результат, а также более согласованная статистическая интерпретация связных по плотности компонент.

DBSCAN – Density-Based Spatial Clustering

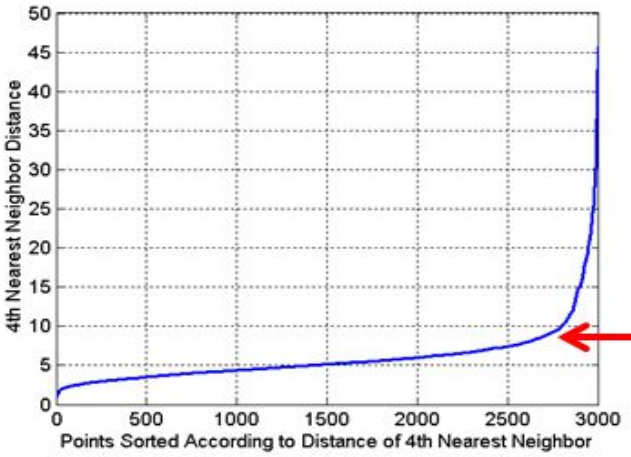
- Качество DBSCAN зависит от измерения расстояния, используемого в функции $\mathrm{regionQuery}\,(P,\varepsilon)$. Наиболее часто используемой метрикой расстояний является евклидова метрика. Особенно для кластеризации данных высокой размерности[en] эта метрика может оказаться почти бесполезной ввиду так называемого «проклятия размерности», что делает трудным делом нахождение подходящего значения $epsilon$ Этот эффект, однако, присутствует в любом другом алгоритме, основанном на евклидовом расстоянии.
- DBSCAN не может хорошо кластеризовать наборы данных с большой разницей в плотности, поскольку не удается выбрать приемлемую для всех кластеров комбинацию $epsilon$ Если данные и масштаб не вполне хорошо поняты, выбор осмысленного порога расстояния epsilon может оказаться трудным.

DBSCAN – Density-Based Spatial Clustering



DBSCAN – Density-Based Spatial Clustering

- Idea is that for points in a cluster, their k^{th} nearest neighbors are at roughly the same distance
- Noise points have the k^{th} nearest neighbor at farther distance
- So, plot sorted distance of every point to its k^{th} nearest neighbor
- Find the distance d where there is a "knee" in the curve
  - Eps = d, MinPts = k



Eps ~ 7-10
MinPts = 4

Критерии качества кластеризации

$$F_0 = \frac{\Sigma_{i<j}\left[y_i = y_j\right]\rho(x_i, x_j)}{\Sigma_{i<j}\left[y_i = y_j\right]}$$

Среднее внутрикластерное расстояние

$$F_1 = \frac{\Sigma_{i<j}\left[y_i \neq y_j\right]\rho(x_i, x_j)}{\Sigma_{i<j}\left[y_i \neq y_j\right]}$$

Среднее межкластерное расстояние

$$F = \sum_{j=1}^{K}\sum_{i=1}^{n}\frac{w_{ij}^2}{n}$$

Коэффициент разбиения

$$H = \sum_{j=1}^{K}\sum_{i=1}^{n}\frac{w_{ij}^2 \cdot \ln\left(w_{ij}\right)}{n}$$

Индекс чёткости

$$NFI = \frac{nF - 1}{K - 1}, \quad NFI \in [0,1]$$

Энтропия разбиения