DBSCAN

(Density-Based Spatial Clustering of Applications with Noise)

версия 5

Исходная статья

Ester, Kriegel, Sander, Xu
“A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise”
Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, Portland, OR, AAAI Press, pp. 226-231. 1996

С какого текста лучше начать изучение?

https://www.quora.com/What-is-an-intuitive-explanation-of-DBSCAN

Перевод статьи
https://habr.com/ru/post/322034/

Напоминание

Если в столбце содержится нечисловая информация, перекодируем,
чтобы все элементы таблицы данных стали числами.

Предварительно отбираем переменные.

Синонимы:
наблюдение == объект == строка == точка.

Если точки расположены близко,
если между точками малое расстояние,
то считаем объекты похожими.

Идея метода

Кластерами объявим области в пространстве
- с высокой плотностью точек
- связные

Как измеряют плотность точек в DBSCAN

Обозначения
x, ***y*** и ***z*** - точки, наблюдения

dist(x, y) – расстояние между точками ***x*** и ***y***

Определение
Шар радиуса **ε** с центром в точке **x** это множество

$$E(x)=\{t: dist(x,t)\leqslant\varepsilon\}$$

Определение
ε - окрестность точки A — шар радиуса **ε** с центром в точке A

Определение
Плотность в окрестности точки A равна
числу точек в окрестности (в шаре),
деленному на объем окрестности (шара).

Упрощения в модели DBSCAN

1. Центрами **ε -** окрестностей могут быть
   только точки из набора данных

2. **ε** один и тот же для всех точек.
   В ходе кластеризации **ε** не меняется.
   Но может различаться в разных кластеризациях.

2.1. ***Следствие.*** Все шары одного объема

3. Плотность может быть либо высокой,
   либо низкой

3. ***Следствие.*** Плотность не вычисляем. Плотности сравниваем.

Определение
В **ε -** окрестности высокая плотность точек,
если в нее попало не менее **m** точек.
Иначе плотность низкая.

m - пороговое значение.
Задается аналитиком.
В scikit-learn центр окрестности включают в подсчет.

Основные параметры процедуры DBSCAN

Надо настроить параметры **ε** и **m** .

ε Если расстояние между точками меньше **ε** ,
точки считаются близкими

m Если в окрестности точки **x**
m или больше точек,
то точка **x** находится в области высокой плотности.

Формула для подсчета расстояний между точками - третий параметр
он формализует, что значит «похожи»

Определение

Точки из **ε**- окрестности объекта **x** будем называть соседями точки **x**

Будем считать их близкими, похожими на **x** точками

Три типа точек в DBSCAN

корневые, граничные и выбросы.

Тип точки зависит от значений трех параметров.

При изменении параметров тип точки может измениться.

Определение

Точка **x** называется **Корневой** ,

если ее **ε-**окрестность содержит не менее **m** объектов:

$|E(x)| \geq m$.

Комментарий 1

Вокруг корневого объекта **x** большая плотность точек.

Корневые точки **x** в ядре кластера

Комментарий 2

Синонимы:

корневая точка == ядерная точка == точка высокой плотности

Определение

Если точка не является корневой,

но среди ее соседей есть корневая точка,

то точка называется **граничной**

<u>Алгоритм кластеризации</u>

Шаг 1 Кластеризация корневых точек (и только корневых)

Составим список объектов из первого кластера

Выберем какой-нибудь корневой объект **p** из набора данных,

Начнем составлять список объектов кластера с него.

Добавим в список всех его соседей.

Продолжим пополнять список объектов кластера.

Для этого начнем перебирать объекты из списка.

Если в ходе перебора встречаем корневую точку, добавляем всех её соседей в кластер.

Каждая точка входит в список один раз, повторений не допускаем.

Когда перебрали все объекты из списка, пополнение невозможно, все точки из списка составляют кластер.

Если в список попали не все точки из набора данных, можно начать составлять новый список, начиная с какого-нибудь другого корневого объекта.

Объекты из нового списка составят следующий кластер.

Новый кластер не будет пересекаться с предыдущим.

Переходим к составлению списка объектов из второго кластера

…

Шаг 2 Распределение граничных точек по кластерам

Шаг 3 Кластер из выбросов

Определение

Если точка не является корневой,

и среди ее соседей нет корневых точек,

то точка называется **выбросом** или **точкой фона**

Кодом **-1** обозначаются наблюдения вне кластеров — шум, выбросы, фон

Анонс
Если все три параметра заданы,
то число кластеров определяется однозначно.
Но состав кластеров может несущественно меняться.

Число кластеров определять не надо.
Но не спешите радоваться.
Вместо определения числа кластеров надо определять **ε** и **m.**
Это гораздо сложнее

Вопросы

1 Верно ли что кластер состоит только из корневых точек?
Если нет, привести пример такого кластера.

2 Могут ли все точки кластера быть корневыми?
Если да, привести пример такого кластера.

3 Могут ли все точки кластера быть выбросами?
Если да, привести пример такого кластера.

4 Могут ли все точки кластера быть граничными?
Если да, привести пример такого кластера.

5 Может ли DBScan выявлять ленточные кластеры?

6 Может ли DBScan выявлять плотные шаровые кластеры?

7 Зависит ли результат кластеризации
от порядка перебора корневых точек?

8 Важен ли способ подсчета расстояний между точками?

9 Какие расстояния мы можем использовать?

10 Может ли шар на плоскости визуально выглядеть как квадрат?

Расстояния между объектами.
Какие варианты уже реализованы в Python.

Список уже реализованных расстояний можно посмотреть в
https://scikit-learn.org/stable/modules/generated/sklearn.metrics.pairwise_distances.html#sklearn.metrics.pairwise_distances

- scikit-learn:

  ['cityblock', 'cosine', 'euclidean', 'l1', 'l2', 'manhattan']. ????

- scipy.spatial.distance:

  ['braycurtis', 'canberra', 'chebyshev', 'correlation', 'dice', 'hamming', 'jaccard', 'kulsinski', 'mahalanobis', 'minkowski', 'rogerstanimoto', 'russellrao', 'seuclidean', 'sokalmichener', 'sokalsneath', 'sqeuclidean', 'yule']

Матрица попарных расстояний может быть сосчитана заранее по любой формуле, тогда испрользуем *metric='precomputed'*

Комментарии. Достоинства и недостатки метода.

DBSCAN может выявлять ленточные кластеры.

DBSCAN выигрывает, когда у нас в данных присутствуют кластеры
на фоне равномерно распределенного набора точек

DBSCAN плохо работает, когда шаровые скопления соединены перемычками.

DBSCAN плохо кластеризует наборы данных с большой разницей в плотности,

так как у всех кластеров одна и та же пара параметров **m** и **ε**

DBSCAN похож на обобщение иерархического кластерного анализа,
когда расстояние между кластерами вычисляется методом ближайшего соседа.
Вместо ближайшего соседа используется **m**–й ближайший сосед.

Метод популярный

В 2010 DBSCAN занял 24-е место в Microsoft Academic Search (Наиболее цитируемые статьи по интеллектуальному анализу данных согласно Microsoft academic search)

В 2014 алгоритм получил премию «проверено временем» SIGKDD Test of Time Award. Премия даётся алгоритмам, которые получили существенное внимание в теории и практике на ведущей конференции по интеллектуальному анализу данных KDD

IMHO метод опирается на теорию графов,
поэтому о нем относительно легко писать статьи.

Дискуссия с другими текстами

"Несмотря на простоту и скорость k-means, у него есть одна очень большая проблема — это заранее заданное количество кластеров. В реальном же мире чаще всего нам не известно на сколько групп следует разбить данные"

Ответ. В реальном же мире чаще всего нам не известны значения **m** и **ε**

Проблема с k-means решается ценой увеличения объема вычислений.

"часть данных могут быть вредными выбросами "

k-means тоже находит выбросы, хотя возможно и хуже, чем DBSCAN

"DBSCAN отлично разбивает множество на оптимальное количество кластеров и не учитывает выбросы"

Абсолютно рекламное заявление, совершенно неизвестно, как искать оптимум

"Кластеры, найденные DBSCAN могут иметь любую форму, в отличии от алгоритма k-means (которые предполагают только выпуклую форму)"

Про k-means верно, но иерархический кластерный анализ умеет искать ленточные кластеры

Отсутствие инструментов подбора параметров алгоритма DBSCAN — основной недостаток метода

Подбор параметров (Из Википедии)

Вариант 1

Перебираем параметры на решетке. Сравниваем значения «силуэта».

Недостаток метода. «силуэт» неприменим для измерения качества ленточных кластеров.

Вариант 2

m:

Если m = 2 получаем иерархический кластерный анализ примененный с методом ближайшего соседа.

Несколько эмпирических правил

m >= 3.

m >= D+1.

m >= 2*D

m = ln(n)

ε :

Если **ε** выбрана слишком малыми, большая часть данных будет отнесена к выбросам, а для слишком больших значений **ε** кластеры будут сливаться и большинство объектов окажутся в одном кластере.

Значение **ε** может быть выбрано с помощью графика, похожего на каменистую осыпь. Упорядочиваем расстояния до m-го ближайшего соседа в возрастающем порядке. Искомое значение ε соответствует «излому» графика.

Обычно малые значения **ε** предпочтительнее.

Расстояние:

Расстояние должно отражать наше представление о схожести объектов.

Дальнейшее развитие метода DBSCAN

OPTICS

OPTICS можно рассматривать как обобщение DBSCAN, в котором параметр ε заменяется максимальным значением, наиболее воздействущим на эффективность. MinPts тогда становится минимальным размером кластера. Хотя алгоритм существенно проще в области выбора параметров, чем DBSCAN, его результаты труднее использовать, так как он обычно даёт иерархическую кластеризацию вместо простого разделения, которое даёт DBSCAN.