**1**. Задача кластеризации

В задаче классификации мы имели дело с восстановлением отображения из множества объектов в конечный набор меток классов. При этом классы были зафиксированы заранее, то есть мы с самого начала примерно понимали, какого рода объекты должны относиться к каждому из них, и мы располагали обучающей выборкой с примерами объектов и классов, к которым они относятся. В задаче кластеризации мы тоже разбиваем объекты на конечное множество классов, но у нас нет ни обучающей выборки, ни понимания, какой будет природа этих классов. То, что модель кластеризации какие-то объекты сочла «похожими», отнеся к одному классу, будет новой информацией, «открытием», сделанным этой моделью. Обучающей выборки у нас также не будет: ведь мы не знаем заранее, что за классы получатся (а иногда и сколько их будет). Таким образом, кластеризация — это задача обучения без учителя. Из-за общего сходства постановок задач в литературе кластеризацию иногда называют unsupervised classification.

Методы кластеризации часто применяют, когда фактически нужно решить задачу классификации, но обучающую выборку собрать затруднительно (дорого или долго). При этом валидационную выборку для оценки результатов кластеризации собрать значительно проще, так как для неё требуется меньше примеров. При этом стоит помнить, что точность работы supervised-методов значительно выше. Поэтому, если обучающую выборку всё-таки можно собрать, лучше решать задачу классификации, чем задачу кластеризации.

**2.** Среднее внутрикластерное расстояние

Смысл среднего внутрикластерного расстояния максимально соответствует названию:

Изображение выглядит как текст, Шрифт, рукописный текст, белый

Автоматически созданное описание

Сумма расстояний между точками из одного и того же кластера делится на количество пар точек, принадлежащих к одному кластеру. В приведённой выше формуле пары вида (𝑥𝑖,𝑥𝑖) включены в рассмотрение, чтобы избежать неопределённости 0/0​ в случае, когда в каждом кластере ровно по одному объекту. Однако иногда записывают суммы по 𝑖<𝑗, просто доопределяя 𝐹0 в описанном случае нулём.

Решая задачу кластеризации, мы хотим по возможности получать как можно более кучные кластеры, то есть минимизировать 𝐹0.

В случае если у кластеров есть центры 𝜇𝑘​, часто рассматривается аналогичная метрика — средний квадрат внутрикластерного расстояния:

Изображение выглядит как Шрифт, текст, белый, диаграмма

Автоматически созданное описание

Среднее межкластерное расстояние

Аналогично среднему внутрикластерному расстоянию вводится среднее межкластерное:

Изображение выглядит как текст, Шрифт, белый, линия

Автоматически созданное описание

Среднее межкластерное расстояние, напротив, нужно максимизировать, то есть имеет смысл выделять в разные кластеры наиболее удалённые друг от друга объекты.

Гомогенность

Для измерения следующих метрик (гомогенности, полноты и V-меры) нам уже потребуется разметка выборки. Будем обозначать кластеры, к которым наш алгоритм кластеризации относит каждый объект, буквами 𝑘∈{1,...,𝐾}, а классы, к которым объекты отнесены разметкой, — буквами с∈{1,...,С}. Разумный вопрос при наличии разметки — зачем нам решать задачу кластеризации, если с разметкой можно поставить задачу как задачу классификации. Это и правда хороший вопрос в том случае, если размеченных данных достаточно много для обучения классификатора. На практике же часто встречаются ситуации, когда данных достаточно для оценки качества кластеризации, но всё ещё не хватает для использования методов обучения с учителем.

Пусть 𝑛 — общее количество объектов в выборке, 𝑛𝑘 — количество объектов в кластере номер 𝑘, 𝑚𝑐​ — количество объектов в классе номер с, а 𝑛𝑐𝑘​ — количество объектов из класса 𝑐*c* в кластере 𝑘. Рассмотрим следующие величины:

Изображение выглядит как текст, Шрифт, рукописный текст, диаграмма

Автоматически созданное описание

Изображение выглядит как текст, чек, снимок экрана, Шрифт

Автоматически созданное описание

показывает, во сколько раз энтропия изменяется за счёт того, что мы считаем известной принадлежность объектов к выделенным нашим алгоритмом кластерам. Худший случай — когда отношение оказывается равным единице (энтропия не изменилась, условная энтропия совпала с обычной), лучший — когда каждый кластер содержит элементы только одного класса и номер кластера, таким образом, точно определяет номер класса (в этом случае ℎ=1).

Тривиальный способ максимизировать гомогенность кластеризации — выделить каждый объект выборки в отдельный кластер.

Полнота

Изображение выглядит как текст, Шрифт, снимок экрана, чек

Автоматически созданное описание

V-мера

Гомогенность и полнота кластеризации – это в некотором смысле аналоги точности и полноты классификации. Аналог F-меры для задачи кластеризации тоже есть, он называется V-мерой и связан с гомогенностью и полнотой той же формулой, что и F-мера с точностью и полнотой:

Изображение выглядит как текст, Шрифт, чек, снимок экрана

Автоматически созданное описание

Коэффициент силуэта

Ещё одна метрика кластеризации, на этот раз уже не требующая разметки, это коэффициент силуэта (silhouette coefficient). Изначально коэффициент определяется для каждого объекта выборки, а метрика для результатов кластеризации всей выборки вводится как средний коэффициент силуэта для всех объектов выборки.

Чтобы ввести коэффициент силуэта 𝑆(𝑥𝑖), нам потребуются две вспомогательные величины. Первая, 𝐴(𝑥𝑖), — это среднее расстояние между 𝑥𝑖​ и объектами того же кластера. Вторая, 𝐵(𝑥𝑖), — это среднее расстояние между 𝑥𝑖​ и объектами следующего ближайшего кластера. Коэффициент силуэта вводится следующим образом:

Изображение выглядит как текст, Шрифт, снимок экрана, документ

Автоматически созданное описание

Коэффициент силуэта особенно полезен (по сравнению с другими приведёнными метриками) тем, что одновременно и не требует разметки, и позволяет подбирать количество кластеров. См. подробнее [в примере из документации scikit-learn](https://scikit-learn.org/stable/auto_examples/cluster/plot_kmeans_silhouette_analysis.html).

**3**. **1. K-means (k-средних)**

* **Как работает**: K-means делит данные на kkk кластеров, где kkk — это заранее заданное количество. Алгоритм работает итеративно:
  1. Выбираются kkk начальных центроидов.
  2. Каждая точка данных назначается ближайшему центроиду.
  3. Пересчитываются центроиды как среднее значение точек в каждом кластере.
  4. Процесс повторяется, пока центроиды не стабилизируются.
* **Плюсы**:
  1. Простота и скорость работы.
  2. Хорошо работает на больших и плотных наборах данных.
* **Минусы**:
  1. Требует заранее знать количество кластеров (kkk).
  2. Чувствителен к выбросам и шуму.
  3. Предполагает, что кластеры имеют сферическую форму.

**2. DBSCAN (Density-Based Spatial Clustering of Applications with Noise)**

* **Как работает**: DBSCAN основан на плотности точек данных:
  1. Точки, находящиеся в областях с высокой плотностью, объединяются в кластеры.
  2. Точки, которые не входят в плотные области, считаются шумом (выбросами).
  3. Кластер формируется, если вокруг точки находится больше заданного числа соседей (ϵ\epsilonϵ-окрестность).
* **Плюсы**:
  1. Не требует заранее задавать количество кластеров.
  2. Способен находить кластеры произвольной формы.
  3. Устойчив к шуму (может помечать точки как выбросы).
* **Минусы**:
  1. Параметры (ϵ\epsilonϵ и минимальное количество точек) чувствительны к выбору.
  2. Плохо работает на данных с переменной плотностью кластеров.

**3. Hierarchical Clustering (Иерархическая кластеризация)**

* **Как работает**: Этот метод строит иерархию кластеров:
  + **Агломеративная кластеризация** (снизу вверх): начинается с того, что каждый объект — это отдельный кластер. Затем кластеры объединяются на основе близости, пока не останется один большой кластер.
  + **Дивизивная кластеризация** (сверху вниз): начинается с одного кластера, содержащего все данные, который затем делится на более мелкие кластеры.
* **Плюсы**:
  + Не требует заранее знать количество кластеров.
  + Позволяет построить дендрограмму, показывающую иерархию кластеров.
* **Минусы**:
  + Высокая вычислительная сложность, особенно на больших наборах данных.
  + Чувствительность к выбросам и шуму.

**4. Gaussian Mixture Models (GMM)**

* **Как работает**: GMM предполагает, что данные представляют собой смесь нескольких гауссовых распределений. Алгоритм оценивает параметры этих распределений (средние, ковариационные матрицы) и находит вероятности принадлежности точек данных к каждому кластеру.
  + Использует метод **EM (Expectation-Maximization)** для оценки параметров модели.
* **Плюсы**:
  + Способен находить кластеры, которые не имеют сферической формы.
  + Возвращает вероятности принадлежности к кластерам.
* **Минусы**:
  + Чувствителен к выбросам и требует задания количества кластеров.
  + Может медленно сходиться на больших наборах данных.

**5. Mean-Shift (Среднесдвиговая кластеризация)**

* **Как работает**: Mean-Shift пытается найти области с высокой плотностью данных. Алгоритм вычисляет **центры плотности** и перемещает точки к более плотным областям (сдвигает их к моде). Процесс продолжается до тех пор, пока все центры плотности не будут найдены.
* **Плюсы**:
  + Не требует задания количества кластеров.
  + Хорошо находит кластеры произвольной формы.
* **Минусы**:
  + Параметр "ширина ядра" (bandwidthbandwidthbandwidth) сильно влияет на результат.
  + Высокая вычислительная сложность.

**6. Agglomerative Clustering (Агломеративная кластеризация)**

* **Как работает**: Это подтип иерархической кластеризации, который начинает с индивидуальных точек и последовательно объединяет их в кластеры на основе близости. При каждом шаге кластеры с наибольшей похожестью объединяются, пока не будет достигнуто заданное количество кластеров.
* **Плюсы**:
  + Не требует задания количества кластеров заранее.
  + Прост в реализации.
* **Минусы**:
  + Высокая вычислительная сложность.
  + Чувствителен к выбросам.

**7. OPTICS (Ordering Points To Identify the Clustering Structure)**

* **Как работает**: Этот алгоритм схож с DBSCAN, но более гибок. OPTICS упорядочивает точки данных, чтобы можно было идентифицировать кластеры различной плотности.
* **Плюсы**:
  + Способен находить кластеры разной плотности.
  + Не требует задания чёткого значения ϵ\epsilonϵ.
* **Минусы**:
  + Более сложен в реализации и медленнее, чем DBSCAN.
  + Результаты сложно интерпретировать.

**8. Birch (Balanced Iterative Reducing and Clustering using Hierarchies)**

* **Как работает**: Birch создаёт небольшие кластеры, называемые **кластерными особенностями (CF)**, и иерархически агрегирует их, что делает его пригодным для кластеризации больших наборов данных.
* **Плюсы**:
  + Эффективен на больших данных.
  + Может быть использован как предварительный шаг для других методов кластеризации.
* **Минусы**:
  + Меньшая точность на данных с низкой плотностью.

**Заключение:**

Каждый алгоритм кластеризации имеет свои сильные и слабые стороны. Выбор метода зависит от структуры данных, целей задачи и вычислительных ограничений.