2021

Лекция 11. Кластеризация

МЕТОДЫ МАШИННОГО ОБУЧЕНИЯ

ПАПУЛИН С.Ю. (papulin.study@yandex.ru)

Дополнительные материалы доступны по ссылке: https://github.com/mlmethods

Содержание

1. Кластеризация

Кластеризация используется для поиска подгрупп, так называемых кластеров, в данных. Когда мы кластеризуем данные, мы разделяем их на отдельные группы так, чтобы наблюдения в рамках одной группы были похожу между собой, в то время как наблюдения из разных групп достаточно отличались бы друг от друга. При этом нам необходимо определить, как выражать сходство и отличие двух и более наблюдений.

Можно сказать, что при кластеризации мы ищем гомогенные подгруппы наблюдений.

Данная задача относится к классу обучения без учителя, так как мы пытаемся раскрыть структуру (в данном случае выявить кластеры) на основе имеющихся данных без целевого значения y.

Далее рассмотрим 4 метода кластеризации:

- Метод k-средних: для заданного количества кластеров определяется их центры в многомерном пространстве признаков.
- Иерархическая кластеризация: в данном случае строится дерево наблюдений, дендрограмма; с её помощью можно представить деление данных на количество кластеров от 1 до n, где n – количество наблюдений.
- DBSCAN: кластеры формируются на основе близости (плотности) между наблюдениями.
- Смесь функций нормального распределения: задается количество кластеров и для каждого из них на основе имеющихся наблюдений подбираются параметры функции нормального распределения.

Пример в маркетинге

Пусть у нас есть большой набор данных, определяющих большое количество людей по множеству признаков (например, медианный доход на домохозяйство, профессия, расстояние до ближайшего городского района и пр.). Наша цель – выполнить сегментацию рынка, определив подгруппы «похожих» клиентов. Если некоторым клиентам из определенного кластера нравится какой-то товар, вероятно он понравится и другим «похожим» клиентам в рамках рассматриваемого кластера. Таким образом можно формировать группы, например, более восприимчивых к определенным формам рекламы или вероятнее купят определенный товар.

В итоге задача сегментации рынка сводится к кластеризации клиентов на основе имеющегося набора данных X без целевого значения y.

2. Метод k-средних

Метод k-средних – простой подход к разделению данных на множество K различных непересекающихся кластеров. Для выполнения кластеризации необходимо задать желаемое количество кластеров K. Затем алгоритм на каждой итерации будет назначать каждому наблюдению один из K кластеров до тех пор, пока не будет выполнен критерий остановки.

Ниже приведен пример назначения разного количества кластеров для одного набора данных.

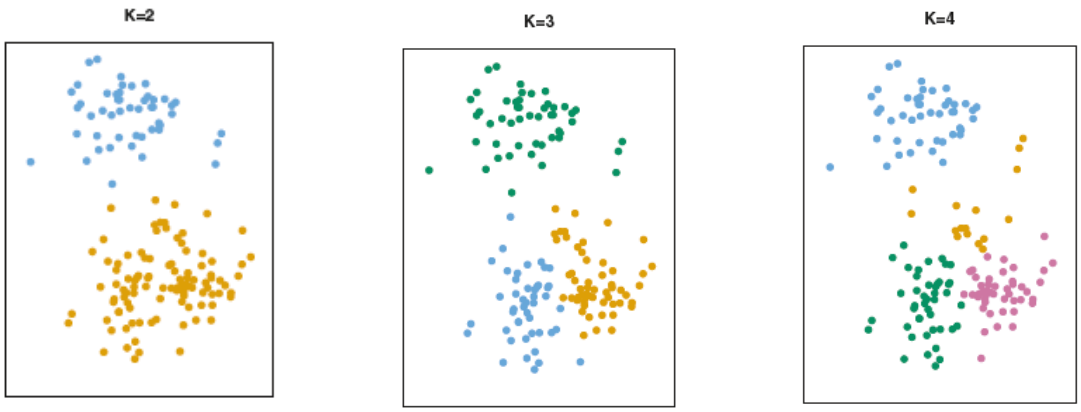


Рисунок – Пример кластеризации с различным количеством кластеров [1]

Представим более формально особенности метода k-средних. Пусть $C_1, \dots, C_K$ обозначают множества, содержащие индексы наблюдений в каждом кластере. Эти множества должны удовлетворять двум свойствам:

- $C_1 \cup C_2 \cup \cdots \cup C_K = \{1, \dots, n\}$: каждое наблюдение принадлежит по крайней мере одному из K кластеров.
- $C_k \cap C_{k'} = \emptyset$ для всех $k \neq k'$: кластеры являются непересекающимися, т. е. наблюдение не может принадлежать более чем одному кластеру.

Если i-ое наблюдение находится в k-ом кластере, то будем обозначать это как $i \in C_k$.

Основная идея, лежащая в основе метода k-средних, является то, что хорошая кластеризация это та, в которой внутрикластерная вариация (т. е. отличие одних наблюдений от других) минимальна. Обозначим меру внутрикластерной вариации как $W(C_k)$ для кластера C_k. Тогда задача сводится к определению

$$C^* = \underset{C_1,\dots,C_K}{\operatorname{argmin}} \sum_{k=1}^{K} W(C_k),$$

где

$$C^* = (C_1^*, \dots, C_k^*).$$

Если взять квадрат Евклидова расстояния в качестве определения близости наблюдений, то $W(C_k)$ можно записать следующим образом

$$W(C_k) = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} \left(x_{ij} - x_{i'j}\right)^2,$$

где $|C_k|$ определяет количество наблюдений для k-го кластера.

Таким образом, внитрикластерная вариация для k-го кластера в данном случае определяется как сумма всех пар квадратных расстояний между наблюдениями k-го кластера, деленное на общее количество наблюдений в этом кластере. Получаем, что необходимо решить следующую задачу минимизации:

$$C^* = \underset{C_1,\dots,C_K}{\operatorname{argmin}} \sum_{k=1}^{K} \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} \left(x_{ij} - x_{i'j}\right)^2$$

Если имеем n наблюдений и K кластеров, то существует почти K^n вариантов того, как мы можем сгруппировать данные. Соответственно, метод k-средних пытается решить данную задачу за меньшее количество шагов. Для этого мы можем преобразовать выражения для $W(C_k)$ следующим образом

$$W(C_k) = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} \left(x_{ij} - x_{i'j}\right)^2 = 2 \sum_{i \in C_k} \sum_{j=1}^{p} \left(x_{ij} - \overline{x}_{kj}\right)^2$$

где

$$\overline{x}_{kj} = \frac{1}{|C_k|} \sum_{i \in C_k} x_{ij},$$

т. е. среднее значение для *j*-го признака в кластере C_k.

В итоге задача минимизации представляется как

$$C^* = \underset{C_1,\ldots,C_K}{\operatorname{argmin}} \sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j=1}^{p} \left(x_{ij} - \overline{x}_{kj}\right)^2.$$

Ниже представлен алгоритм k-средних, позволяющий найти локальный минимум для обозначенной задачи.

Алгоритм

1. Случайным образом присваиваем значение от 1 до K каждому наблюдению
2. Повторяем до тех пор, пока не будет изменяться наблюдения в кластерах
   - а. Для каждого из K кластеров вычисляем центроид кластера. Центроид для k кластера есть вектор средних значений размера p по каждому признаку наблюдений данного кластера.

     Центроид кластера:

     $$\overline{\boldsymbol{x}}_k = \left[\overline{x}_{k1}, \overline{x}_{k2}, \ldots, \overline{x}_{kp}\right]^T,$$

     где

     $$\overline{x}_{kj} = \frac{1}{|C_k|} \sum_{i \in C_k} x_{ij}$$

   - б. Присваиваем каждому наблюдению кластер, чей центроид ближе (по Евклидову расстоянию). Получаем обновленный состав кластеров $C_1, \ldots, C_K$.

     Расстояние от точки $\boldsymbol{x}_i$ до центра кластера $\overline{\boldsymbol{x}}_k$:

     $$d(\boldsymbol{x}_i, \overline{\boldsymbol{x}}_k) = \sum_{j=1}^{p} \left(x_{ij} - \overline{x}_{kj}\right)^2$$

     Ближайший кластер $\hat{y}_i$ для точки $\boldsymbol{x}_i$:

     $$\hat{y}_i = \underset{k \in \{1,\ldots,K\}}{\operatorname{argmin}} d(\boldsymbol{x}_i, \overline{\boldsymbol{x}}_k)$$

В процессе работы алгоритма мы постоянно уменьшаем внутрикластерную вариацию до тех пор, пока не перестанут происходить изменения и, соответственно, целевая функция (в обозначенной ранее задаче оптимизации) не будет увеличиваться. В этом случае мы говорим о том, что достигли локального минимума.

Пример

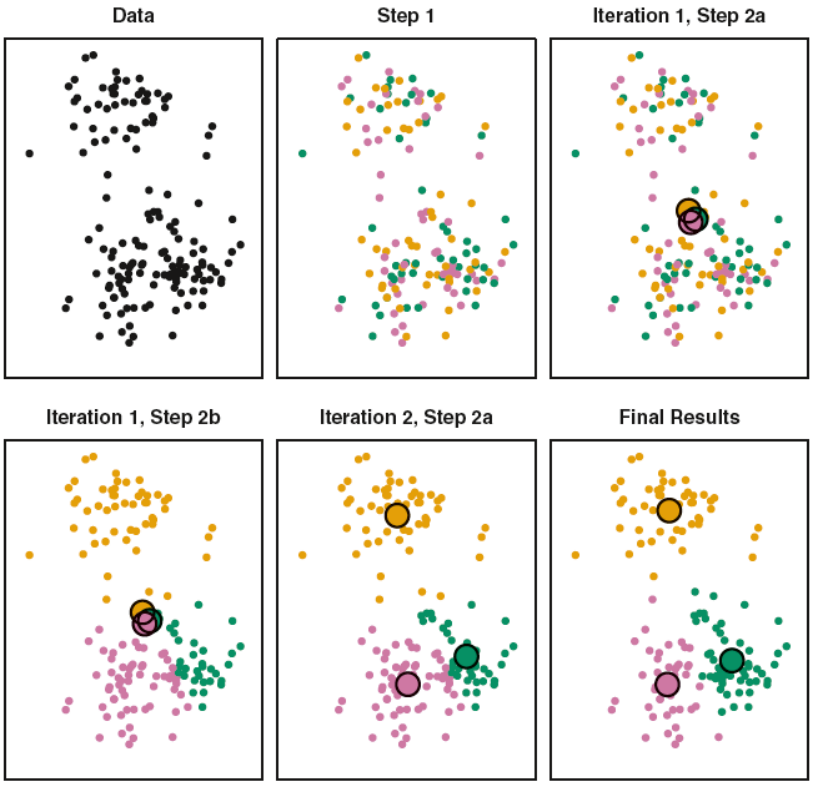


Рисунок – Пример итераций в методе k-средних [1]

Так как алгоритм k-средних ищет локальный, а не глобальный минимум, получаемый результат будет зависеть от начального назначения кластеров для каждого наблюдения, которое происходит случайным образом. Поэтому важно запустить алгоритм несколько раз и выбрать то решение, при котором целевая функция будет иметь наименьшее значение.

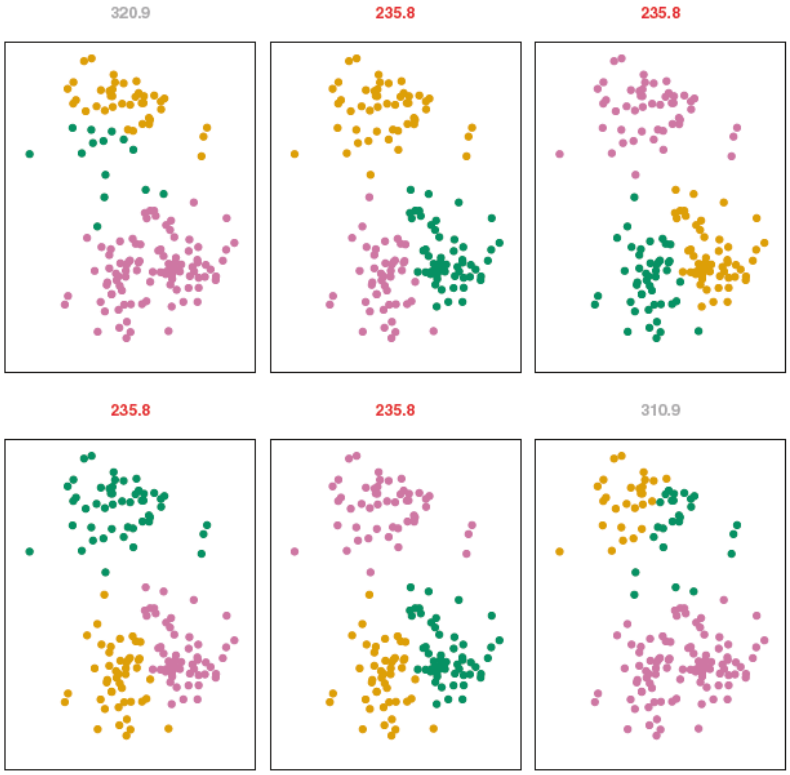


Рисунок – Пример сходимости алгоритма k-средних при 6 запусках [1]

Также при использовании k-средних следует решить, стандартизовать признаки или нет. Стандартизация может быть полезной, например, в случаях с разными единицами измерения.

В целом к особенностям метода k-средних можно отнести следующие:

- необходимость заранее знать количество кластеров
- выбор способа задания начальных кластеров для наблюдений
- каждое наблюдение принадлежит только одному кластеру
- необходимо определить сколько раз запускать алгоритм

В качестве достоинств можно выделить простоту, сходимость, применимость к большим наборам данных, а к недостаткам можно отнести необходимость заранее знать количество кластеров, зависимость от начального задания кластеров, чувствительность к выбросам, плохо ведет себя при большой размерности данных, не учитывает плотность и размеры кластеров

3. Иерархическая кластеризация

3.1. Общие сведения

В качестве недостатка метода k-средних можно выделить необходимость при запуске алгоритма задавать конечное количество кластеров K. Иерархическая кластеризация – альтернативный подход, который в общем виде этого не требует.

Агломеративная кластеризация – наиболее распространенный тип иерархической кластеризации, при которой строится древообразная структура (дендрограмма) начиная с

листьев до основания (ствола) дерева, постепенно объединяя наблюдения в кластеры до тех пор, пока все они не будут принадлежать одному кластеру.

Таким образом, при движении вверх по дереву ветви объединяются с листьями или с другими ветвями. Чем раньше происходит объединение, тем больше похожи наблюдения в рамках каждой группу. В то же время наблюдения, которые объединяются на более поздних этапах, могут весьма отличаться друг от друга. В целом для исходных данных существует 2^{n-1} возможных вариантов дендрограммы, где n есть количество листьев.

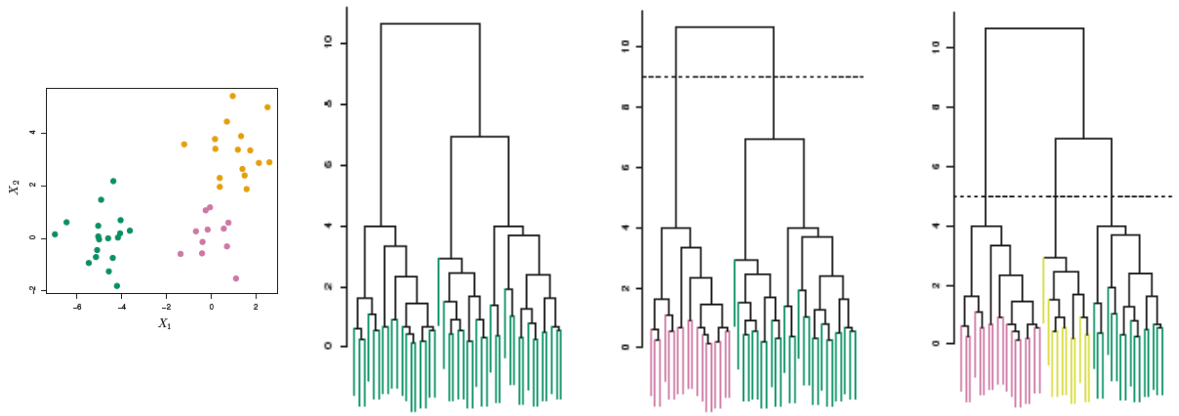


Рисунок – Пример дендрограммы с разным уровнем отсечения [1]

Для любых двух наблюдений мы можем найти точку, в которой ветви, содержащие эти наблюдения, впервые объединяются. Высота слияния, измеряемое по вертикальной оси, указывает на то, как сильно отличаются эти два наблюдения. Таким образом, простроив дендрограмму, мы можем произвести кластеризацию наблюдения на основе значения точки слияния.

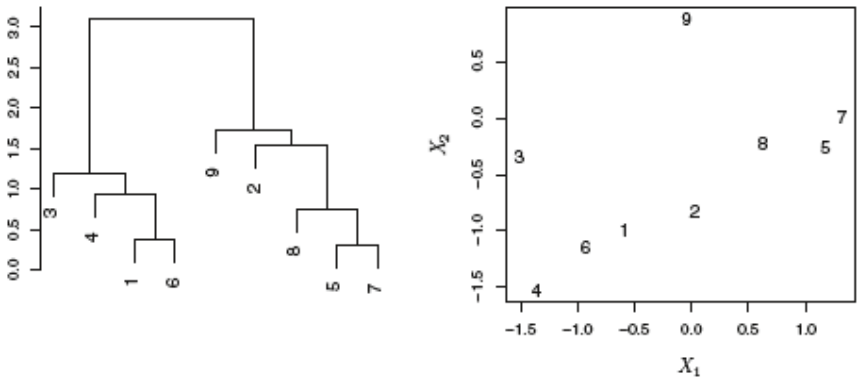


Рисунок – Пояснение к высоте слияния [1]:

- В данном случае дендрограмма строится с использованием Евклидова расстояния и метода полной связи (complete linkage) (см. далее)
- Наблюдения 5 и 7 достаточно похожи между собой, так же как и наблюдения 1 и 6.
- В то же время наблюдение 9 значительно менее похоже на наблюдение 2, чем наблюдения 8, 5 и 7, даже если наблюдения 9 и 2 близко расположены по горизонтали дендрограммы.

Таким образом, посредством дендрограммы мы не можем сделать вывод о схожести двух наблюдений на основе их близости вдоль горизонтальной оси. Напротив, мы можем говорить о

схожести двух наблюдений по вертикальной оси на основе положения точки, в которой произошло первое слияние ветвей, содержащих эти два наблюдения.

Для определения количества кластеров на основе дендрограммы, мы делаем горизонтальный срез. Различные множества наблюдений ниже среза интерпретируются как кластеры. Контролируя уровень среза по вертикали, мы можем получить любое количество кластеров от 1 (без среза) до n (когда срез на уровне 0, т.е. каждое наблюдение формирует свой собственный кластер).

При увеличении высоты/уровня среза дендрограммы наблюдения либо остаются в прежних кластерах, либо количество кластеров уменьшается. Таким образом получается иерархическая структура по кластерам наблюдений, т. е. кластеры и их наблюдения на более высоком уровне среза будут включать в себя кластеры, полученные при использовании меньшей высоты среза.

Таким образом, высота среза по дендрограме выполняет ту же роль, что и задаваемое количество кластеров в методе k-средних: контролирует количество кластеров.

3.2. Алгоритм построения дендограммы

Особенности алгоритма:

- Алгоритм выполняется итеративно.
- Построение дендрограмы начинается снизу, когда каждое наблюдение представляется в виде отдельного кластера.
- Два наиболее схожих кластера объединяются, таким образом формируется $n-1$ кластер
- Далее следующие два наиболее схожих кластера объединяются, формируя $n-2$ кластера
- Операции повторяются до тех пор, пока все наблюдения не будут принадлежать одному кластеру
- В этом случае построение дендограммы завершается

Алгоритм

1	Определить попарно близость наблюдений $\binom{n}{2} = n(n-1)/2$, то есть между двумя наблюдениями в различных сочетаниях. На начальном этапе каждое наблюдение рассматривать как отдельный кластер
2	Для $i = n, n-1, \dots, 2$
а	Определить наиболее близкие пары кластеров, объединить эти кластеры. Величина близости указывает на высоту в дендрограмме
б	Определить межкластерную близость для всех пар кластеров

Как обозначено в алгоритме, для построения дендрограммы необходимо определить меру близости между кластерами, которая в свои очередь зависит от близости между парами наблюдений двух кластеров. Далее рассмотрим какие могут быть использованы меры близости для пары наблюдений и пары кластеров.

Замечание

Под близостью в данном случае понимается некоторая мера сходства/различая между векторами.

При этом мера сходства (similarity) лежит в диапазоне от 0 (несхожи) до 1 (схожи). Мера различия (dissimilarity) от 0 (нет отличий) до ∞ (или 1).

Например, Евклидово расстояние, если используется в качестве меры близости, является мерой различия, в то время как коэффициент корреляции Пирсона – мера сходства, как и косинусное сходство (cosine similarity).

3.3. Мера близости между двумя наблюдениями

В качестве меры близости используют:

- Евклидово расстояние

$$d_{ii'} = \|\boldsymbol{x}_i - \boldsymbol{x}_{i'}\|$$

- расстояние на основе корреляции (например, Пирсона)

$$d_{ii'} = \frac{\sum_{j=1}^{p}(x_{ij} - \mu_i)(x_{i'j} - \mu_{i'})}{\sqrt{\sum_{j=1}^{p}(x_{ij} - \mu_i)^2}\sqrt{\sum_{j=1}^{p}(x_{i'j} - \mu_{i'})^2}},$$

где μ_i – среднее значение признаков i-ого наблюдение.

Расстояние на основе корреляции рассматривает два наблюдения похожими, если их признаки имеют высокую корреляцию, даже если наблюдаемые значения далеки друг от друга с точки зрения Евклидова расстояния. Данная мера ориентирована на форму наблюдений по их признакам, нежели на их величину. Выбор меры схожести/различия является важным, так как это влияет на результирующую дендрограмму.

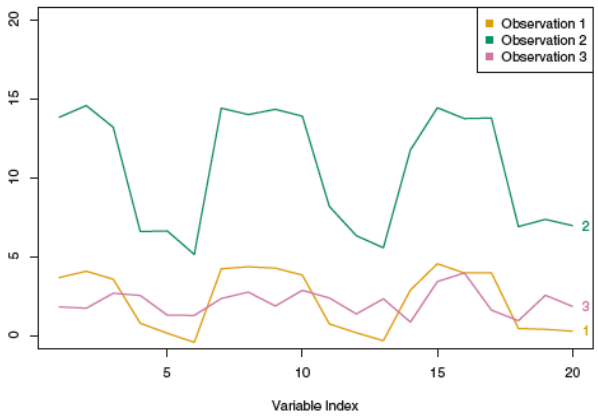


Рисунок – Пример с коррелированными и некоррелировнными по признакам наблюдениями [1]

Пример

Целью является идентификация подгрупп похожих покупателей таких, что покупателям в рамках каждой подгруппы демонстрировались бы товары и реклама, которые вероятно их заинтересуют.

Предположим, что набор данных имеет вид матрицы, в которой строками являются покупатели, столбцами – доступные для покупки товары и значение указывает на количество купленного определённого товара конкретным покупателем (например, если покупатель не покупал товар, то 0, если купил один раз, то 1).

Если используется Евклидово расстояние, то покупатели, которые в целом купили небольшое количество товаров (например, нечастые пользователи онлайн магазинов) будут сгруппированы вместе в один кластер, что может быть нежелательно. С другой стороны, если использовать расстояние на основе корреляции, то покупатели с похожими предпочтениями (например, покупатели, которые купили товары A и B, но не товары C и D) будут объединены в один кластер, даже если покупатели со схожими предпочтениями отличаются по объему покупок. Поэтому расстояние на основе корреляции может быть более правильным выбором для данной задачи.

3.4. Мера близости между двумя кластерами

Концепция близости между парой наблюдений должна быть расширена для пары групп наблюдений, то есть кластеров. Это достигается с использованием такого понятия как связность (linkage), которое определяет сходство/различия двух групп наблюдений.

Общепринятыми типами связности являются:

- метод одиночной связи (single linkage):

$$d_{SL}(G,H) = \min_{\substack{i\in G \\ i'\in H}} d_{ii'},$$

где G и H – два кластера; $d_{ii'}$ – расстояние между двумя наблюдениями разных кластеров;

- метод полной связи (complete linkage):

$$d_{SL}(G,H) = \max_{\substack{i\in G \\ i'\in H}} d_{ii'}$$

- метод средней связи (average linkage):

$$d_{GA}(G,H) = \frac{1}{N_G N_H}\sum_{i\in G}\sum_{i'\in H} d_{ii'},$$

где N_G и N_H – количество наблюдений в кластерах G и H, соответственно;

- центроид (centroid)

Первые три являются наиболее популярными. Методы средней и полной связи в целом предпочтительнее метода одиночной связи, так как с их использованием, как правило, получаются более сбалансированные дендрограммы (см. рисунок ниже). Центроидная связь

часто используется в геномике, но имеет значительный недостаток, связанный с инверсией. Под инверсией подразумевается ситуация, когда два кластера объединяются на высоте ниже каких-либо из его составляющих кластеров. Это приводит к сложности в визуализации и интерпретации дендрограмы.

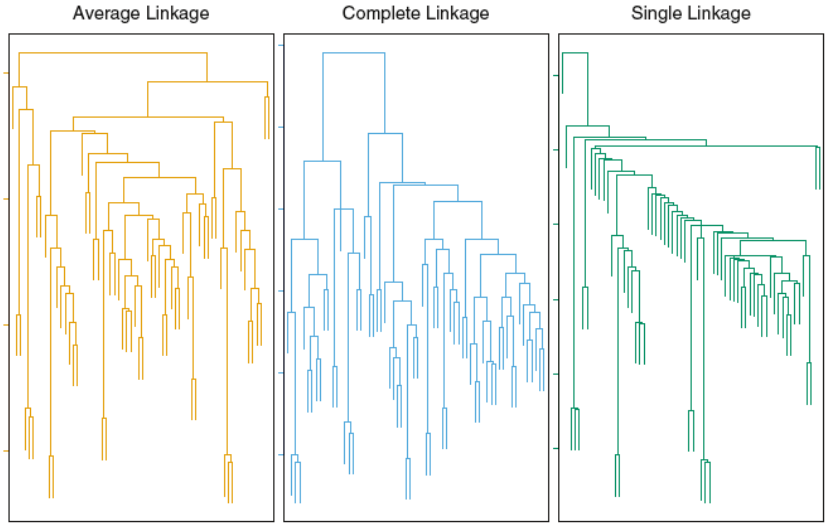


Рисунок – Дендограммы с разным типом связности [1]

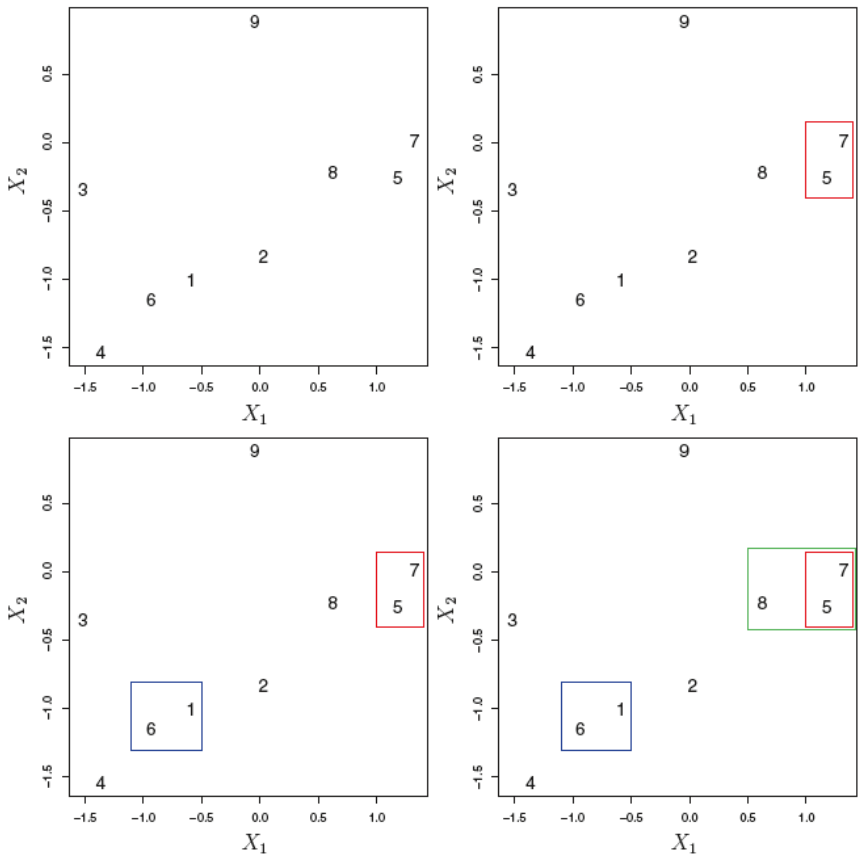


Рисунок – Пример первых нескольких операций построения дендограммы с использованием Евклидова расстояния и полной связи [1]

При использовании иерархической кластеризации необходимо решить, стоит ли использовать стандартизацию признаков или нет.

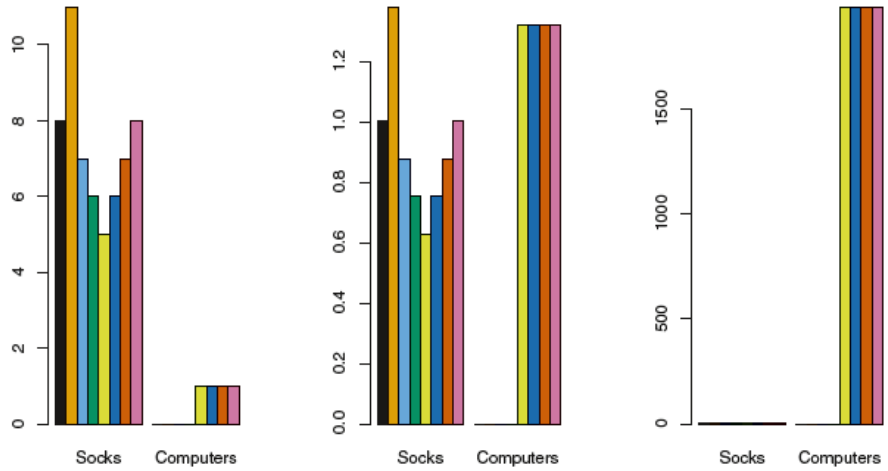


Рисунок – Пример масштабирования признаков (два товара) для 8 покупателей [1]

Особенностью иерархической кластеризации являются:

- Выбор меры схожести/различия между наблюдениями
- Выбор метода связи между группами наблюдений
- Построение дендрограммы
- Выбор высоты среза дендрограммы для получения кластеров

4. Кластеризация на основе плотности (DBSCAN)

Алгоритм подразделят все наблюдения на три категории:

- основные точки (core points)
- неосновные точки (non-core points)
- выбросы (outliers)

Основные и неосновные являются точками кластеров. Для отнесения точек к одной из трёх категорий используются два параметра $n_{\min}$ и ε, которые выступают в роли параметров плотности кластеров. $n_{\min}$ определяет минимальное количество наблюдений в окрестности точки для того, чтобы она рассматривалась как основная (включая саму точку). Окрестность точки контролируется параметром ε, который определяет максимальное расстояние между точками. Таким образом, чтобы точка стала основной, в её окрестности ε должно быть $n_{\min}$ точек. Если точка лежит на расстоянии ε от основной, но при этом количество соседей меньше $n_{\min}$, то в этом случае точка называется неосновной. Все остальные точки рассматриваются как выбросы.

Алгоритм начинает свою работу с выбора случайного наблюдения, определяет количество соседей и относит точку к определённой категории. Далее эта же процедура повторяется для всех выявленных соседей. Все соседние основные и неосновные точки формируют один кластер. Когда все соседние точки размечены, случайным образом выбирается следующая точка и алгоритм пытается построить новый кластер.

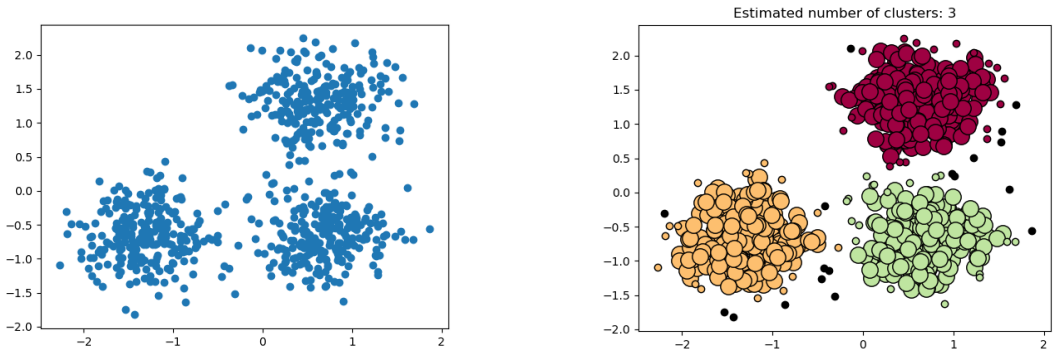


Рисунок – Пример кластеризации с DBSCAN [3]

Преимуществом данного алгоритма является то, что количество кластеров определяется автоматически и кластеры могут иметь произвольную форму. Однако подбор параметров $n_{\min}$ и ε может быть не простой задачей. Так же следует отметить, что алгоритм плохо справляется, когда данные сгруппированы с разной плотностью.

5. Смесь Гауссовских распределений (Gausian Mixture Models)

Ранее, когда рассматривали методы регрессии и классификации, мы часто делали предположение о законе распределения наблюдений, что позволяло нам упростить обоснование и вывод оценок параметров соответствующих моделей предсказания. Однако в реальности данные могут значительно отличаться от упрощенной модели представления, например, нормального закона распределения. Выборка может продемонстрировать, например, что генеральная совокупность следует рассматривать как мультимодальное распределение. В этом случае мы можем сформировать новое распределение на основе более простых. Каждое такое простое распределение будет рассматриваться как компонента смеси (см. рис. ниже).

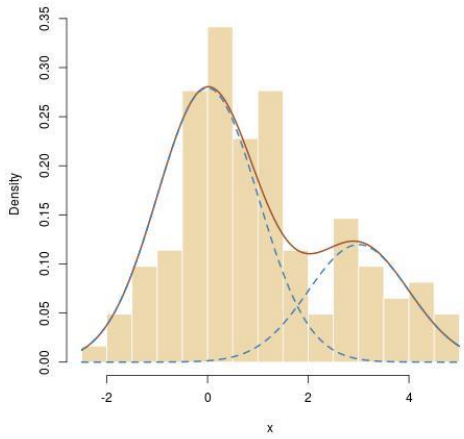


Рисунок – Мультимодальная выборка и её аппроксимация посредством смеси двух функций нормального закона распределения

В контексте задачи кластеризации мы можем представить, что каждая компонента смеси определяет отдельный кластер, и тогда задача сводится к поиску параметров закона распределения каждой компоненты. Далее приводится более подробное описание того, как это можно реализовать с использованием нормального закона распределения.

Так как у нас есть только признаки X и нет целевых значений y, то нельзя сказать заранее сколько всего групп (или кластеров). Однако можно сделать предположение, что данные представляются в виде K компонент (или групп) и каждая компонента описывается нормальным законом распределения (или распределение Гаусса) с параметрами $\boldsymbol{\mu}_k$ и Σ_k. В итоге необходимо найти такие параметры $\boldsymbol{\mu}_k$ и Σ_k для каждой компоненты, при которых с максимальной вероятностью можно получить исходную выборку X. Соответственно, это можно представить в виде максимизации функции правдоподобия

$$\hat{\theta} = \arg\max_{\theta} L(X;\theta),$$

где θ – неизвестные параметры (в данном случае это $\boldsymbol{\mu}_k$, Σ_k и вес компоненты π_k).

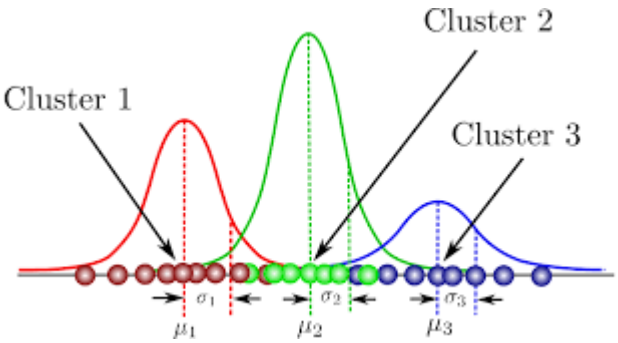


Рисунок – Пример с тремя компонентами [xxx]

Функция правдоподобия можно представить как

$$L(X;\theta) = p(X;\theta) = \prod_{i=1}^{n} p(\boldsymbol{x}_i;\theta)$$

Если учитывать предположение о K компонентах, то вероятность наблюдать $\boldsymbol{x}$ складывается из вероятностей наблюдения $\boldsymbol{x}$ при условии принадлежности его к разным компонентам

$$p(\boldsymbol{x};\theta) = \sum_{k=1}^{K} p(\boldsymbol{x}, y=k;\theta) = \sum_{k=1}^{K} p(y=k;\theta)p(\boldsymbol{x}|y=k;\theta),$$

и

$$\sum_{k=1}^{K} p(y=k;\theta) = 1,$$

где y – скрытая переменная (в данном случае отвечает за кластер и принимает значение от 1 до K)

Замечание

В данном случае у нас непрерывная величины, поэтому речь идет о значении плотности вероятности, а не о вероятности как таковой. Как уже говорилось ранее в лекциях, вероятность наблюдать $\boldsymbol{x}$ для непрерывной величины равна нулю.

Если учитывать, что компоненты представляются нормальным законом распределения, то

$$p(\boldsymbol{x}|y=k;\theta)=\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k,\Sigma_k),$$

где $\boldsymbol{\mu}_k$ – вектор среднего значения кластера для компоненты k; Σ_k – матрица ковариации для компоненты k.

В результате получим, что функция правдоподобия будет иметь следующий вид

$$L(X;\theta)=\prod_{i=1}^{n}\sum_{k=1}^{K}p(y=k;\theta)p(\boldsymbol{x}_i|y=k;\theta)=\prod_{i=1}^{n}\sum_{k=1}^{K}\pi_k\mathcal{N}(\boldsymbol{x}_i|\boldsymbol{\mu}_k,\Sigma_k),$$

где $\pi_k=p(y=k;\theta)$

Чтобы найти оценки неизвестных параметров π_k, $\boldsymbol{\mu}_k$ и Σ_k, можно воспользоваться методом максимального правдоподобия, то есть

$$\underbrace{\widehat{\boldsymbol{\pi}},\widehat{\boldsymbol{\mu}},\widehat{\boldsymbol{\Sigma}}}_{\widehat{\theta}}=\underset{\theta}{\operatorname{argmax}}\,L(X;\theta)=\underset{\theta}{\operatorname{argmax}}\prod_{i=1}^{n}\sum_{k=1}^{K}\pi_k\mathcal{N}(\boldsymbol{x}_i|\boldsymbol{\mu}_k,\Sigma_k),$$

где $\hat{\theta}$ – оценки неизвестных параметров по всем компонентам, $\hat{\theta}=\{\hat{\pi}_k,\widehat{\boldsymbol{\mu}}_k,\widehat{\Sigma}_k\}_{k=1\ldots K}$.

То же выражение можно записать с использованием логарифма

$$\underbrace{\widehat{\boldsymbol{\pi}},\widehat{\boldsymbol{\mu}},\widehat{\boldsymbol{\Sigma}}}_{\widehat{\theta}}=\underset{\theta}{\operatorname{argmax}}\log L(X;\theta)=\underset{\theta}{\operatorname{argmax}}\sum_{i=1}^{n}\log\sum_{k=1}^{K}\pi_k\mathcal{N}(\boldsymbol{x}_i|\boldsymbol{\mu}_k,\Sigma_k)$$

Так как под логарифмом у нас сумма, напрямую оценить значения параметров θ, как мы делали раньше в виде закрытой формы, непростая задача. Однако такую задачу можно решить посредством EM-алгоритма (**E**xpectation-**M**aximization). Следует подчеркнуть, что функция правдоподобия может иметь несколько локальных максимумов, поэтому EM-алгоритм не гарантирует попадания в глобальный максимум.

EM-алгоритм выполняется в два шага с последующим их повторением до остановки по заданному критерию. Ниже приведен более подробная работа алгоритма для нашей задачи кластеризации.

Замечание

Следует отметить, что математическое обоснование EM-алгоритма выходит за рамки обсуждаемой темы, поэтому далее приводятся только конечные выражения, полученные посредством данного алгоритма.

Алгоритм – EM-алгоритм для кластеризации посредством смеси гауссовских моделей

1	Инициализация (случайная или с применением k-средних) значений π_k, $\boldsymbol{\mu}_k$ и Σ_k

2	Цикл
2а	E-шаг
	$r_{i,k} = \frac{\pi_k^{(t)} \mathcal{N}(\boldsymbol{x}_i \mid \boldsymbol{\mu}_k^{(t)}, \Sigma_k^{(t)})}{\sum_{j=1}^{K} \pi_j^{(t)} \mathcal{N}(\boldsymbol{x}_i \mid \boldsymbol{\mu}_j^{(t)}, \Sigma_j^{(t)})}$
2б	M-шаг
	$\pi_k^{(t+1)} = \frac{1}{n} \sum_{i=1}^{n} r_{i,k}$ $\boldsymbol{\mu}_k^{(t+1)} = \frac{1}{\sum_{i=1}^{n} r_{i,k}} \sum_{i=1}^{n} r_{i,k} \, \boldsymbol{x}_i$ $\Sigma_k^{(t+1)} = \frac{1}{\sum_{i=1}^{n} r_{i,k}} \sum_{i=1}^{n} r_{i,k} \left(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{(t+1)}\right)\left(\boldsymbol{x}_i - \boldsymbol{\mu}_k^{(t+1)}\right)^T$
2в	Оценка функции правдоподобия и проверка на сходимость либо по параметрам, либо по функции правдоподобия. Если критерий остановки не соблюдается, то переход на шаг 2а
	$\log L(X; \theta) = \sum_{i=1}^{n} \log \sum_{k=1}^{K} \pi_k \mathcal{N}(\boldsymbol{x}_i \mid \boldsymbol{\mu}_k, \Sigma_k)$

Замечание

E-шаг соответствует случаю, когда $p^{(t)}(y_i = k) = p\left(y_i = k \middle| \boldsymbol{x}_i; \theta^{(t)}\right)$ при фиксированных параметрах $\theta^{(t)}$:

$$p^{(t)}(y_i = k) = p(y_i = k | \boldsymbol{x}_i; \theta^{(t)}) = \frac{p(y_i = k; \theta^{(t)}) p(\boldsymbol{x}_i | y_i = k; \theta^{(t)})}{p(\boldsymbol{x}_i; \theta^{(t)})}$$

M-шаг максимизирует нижнюю границу логарифма функции правдоподобия $Q\left(p^{(t)}(Y), \theta\right)$ при фиксированном $p^{(t)}(Y)$

$$\theta^{(t+1)} = \underset{\theta}{\operatorname{argmax}} \sum_{i=1}^{n} \sum_{k=1}^{K} p^{(t)}(y_i = k) \log p(y_i = k; \theta) p(\boldsymbol{x}_i | y_i = k; \theta) + \text{const}$$

$$\log L\left(X; \theta^{(t)}\right) = Q\left(p^{(t)}(Y), \theta^{(t)}\right) \leq Q\left(p^{(t)}(Y), \theta^{(t+1)}\right) \leq \log L\left(X; \theta^{(t+1)}\right)$$

Далее приравниваем частные производные по параметрам θ к нулю, решаем систему уравнении и получаем новые оценки параметров $\theta^{(t+1)}$.

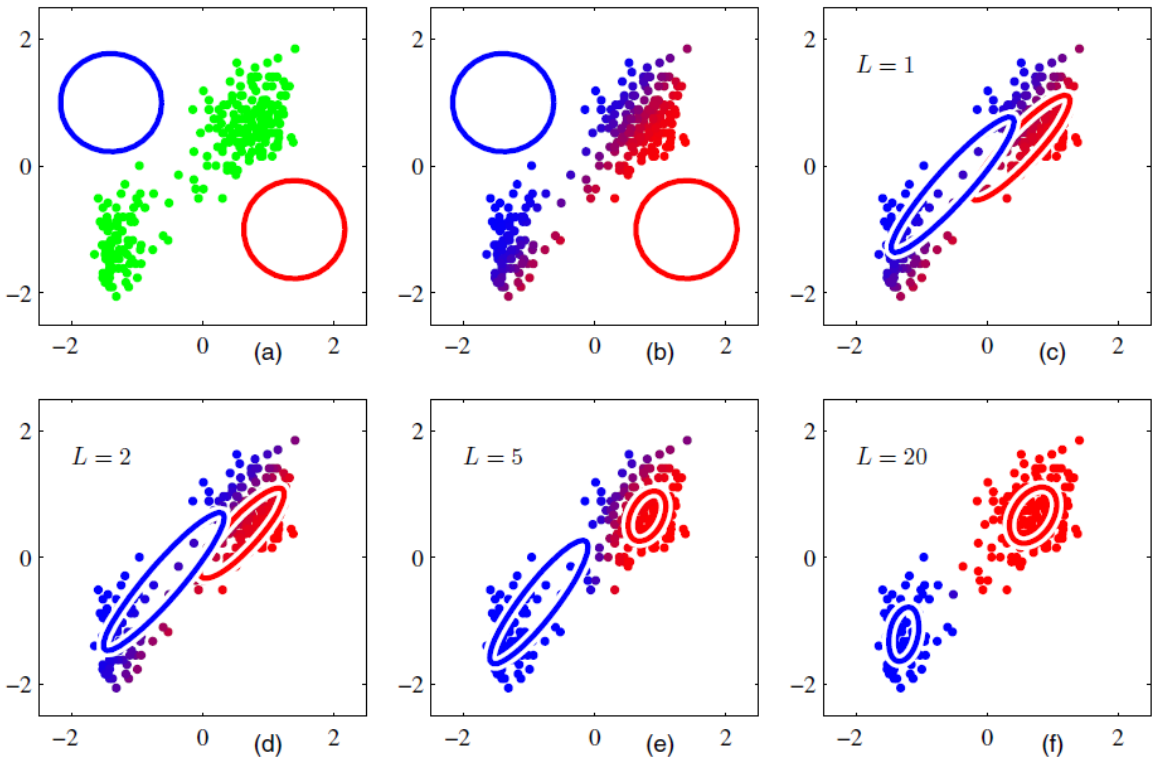


Рисунок – Пример кластеризации посредством смеси гауссовских моделей: оценка апостериорной вероятности (E-шаг) и максимизация для оценки параметров функций нормального закона распределения (M-шаг) [4]

EM для k-средних

Особенностью метода k-средних является то, что каждому наблюдению назначается только один кластер, в то время как гауссовская смесь использует апостериорную вероятность принадлежности наблюдения разным кластерам.

В упрощенном виде k-средних в виде EM шагов можно представить следующим образом.

E-шаг

$$r_{i,k} = \begin{cases} 1, \text{если } k = \underset{j}{\operatorname{argmin}} \left\| \boldsymbol{x}_i - \boldsymbol{\mu}_j^{(t)} \right\|^2 \\ 0 \text{ иначе} \end{cases}$$

M-шаг

$$\boldsymbol{\mu}_k^{(t+1)} = \frac{\sum_{i=1}^{n} r_{i,k} \, \boldsymbol{x}_i}{\sum_{i=1}^{n} r_{i,k}}$$

Связь с EM

Представим ковариационные матрица компонент, как

$$\Sigma_k = \epsilon \mathrm{I},$$

где ϵ – значение дисперсия; I – единичная матрица.

Тогда

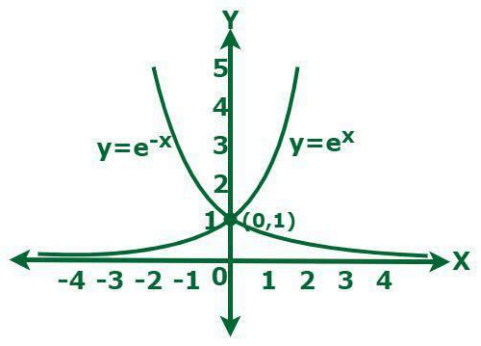
$$p(\boldsymbol{x}|y=k;\theta)=\mathcal{N}(\boldsymbol{x}|\boldsymbol{\mu}_k,\Sigma_k)=\frac{1}{\sqrt{2\pi\epsilon}}\exp\left[-\frac{1}{2\epsilon}\|\boldsymbol{x}-\boldsymbol{\mu}_k\|^2\right],$$

В данном случае ϵ рассматривается как фиксированное значение

Е-шаг для k-средних:

$$r_{i,k}=\frac{\pi_k^{(t)}\exp\left[-\frac{1}{2\epsilon}\left\|\boldsymbol{x}_i-\boldsymbol{\mu}_k^{(t)}\right\|^2\right]}{\sum_{j=1}^{K}\pi_j^{(t)}\exp\left[-\frac{1}{2\epsilon}\left\|\boldsymbol{x}_i-\boldsymbol{\mu}_j^{(t)}\right\|^2\right]}$$

Если $\epsilon\to 0$, то компонент с наименьшим значением $\left\|\boldsymbol{x}_i-\boldsymbol{\mu}_k^{(t)}\right\|^2$ будет медленнее стремиться к нулю, чем остальные. Получим, что для каждого наблюдения $\boldsymbol{x}_i$ все экспоненты будут равны нулю кроме одной. Из этого следует, что все значения $r_{i,k}$ будут равны нулю кроме одного, которое будет равно единице.



Таким образом, получается «жесткое» назначение кластера каждому наблюдению. Следует отметить, что k-средних не оценивает Σ_k, только средние значения $\boldsymbol{\mu}_k$.

Ниже представлен пример кластеризации методом k-средних.

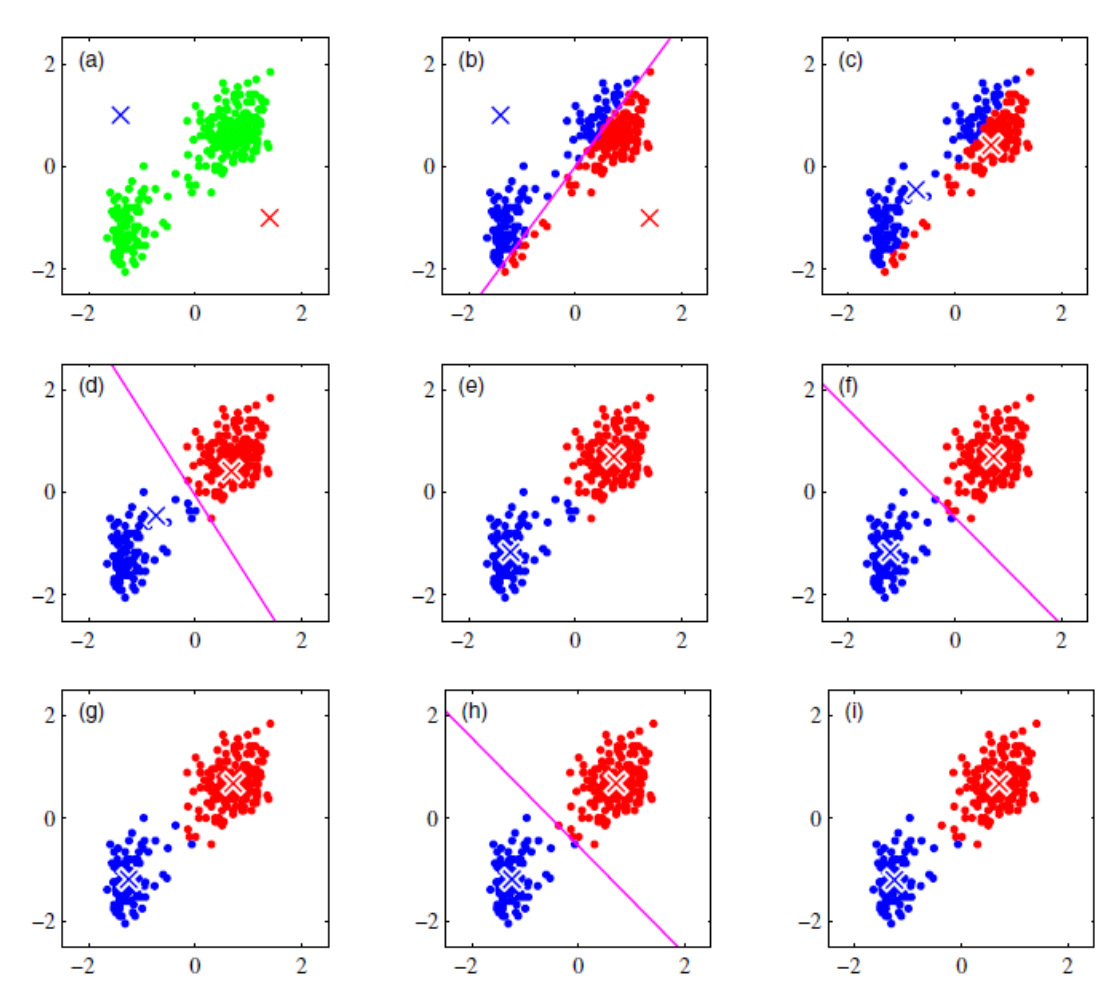


Рисунок – Пример кластеризации посредством k-средних [4]

Ниже приведены сводная таблица по рассмотренным методам кластеризации и результаты их работы для различных наборов данных.

Особенность метода	K-средних	Иерархическая	DBSCAN	GM
Количество кластеров	+	-	-	+
Большое количество наблюдений	+	+	+	-
Большое количество признаков	+/-	+	+/-	-
Произвольная форма кластеров	-	+/-	+	+/-
Предсказание	+	-	-	+

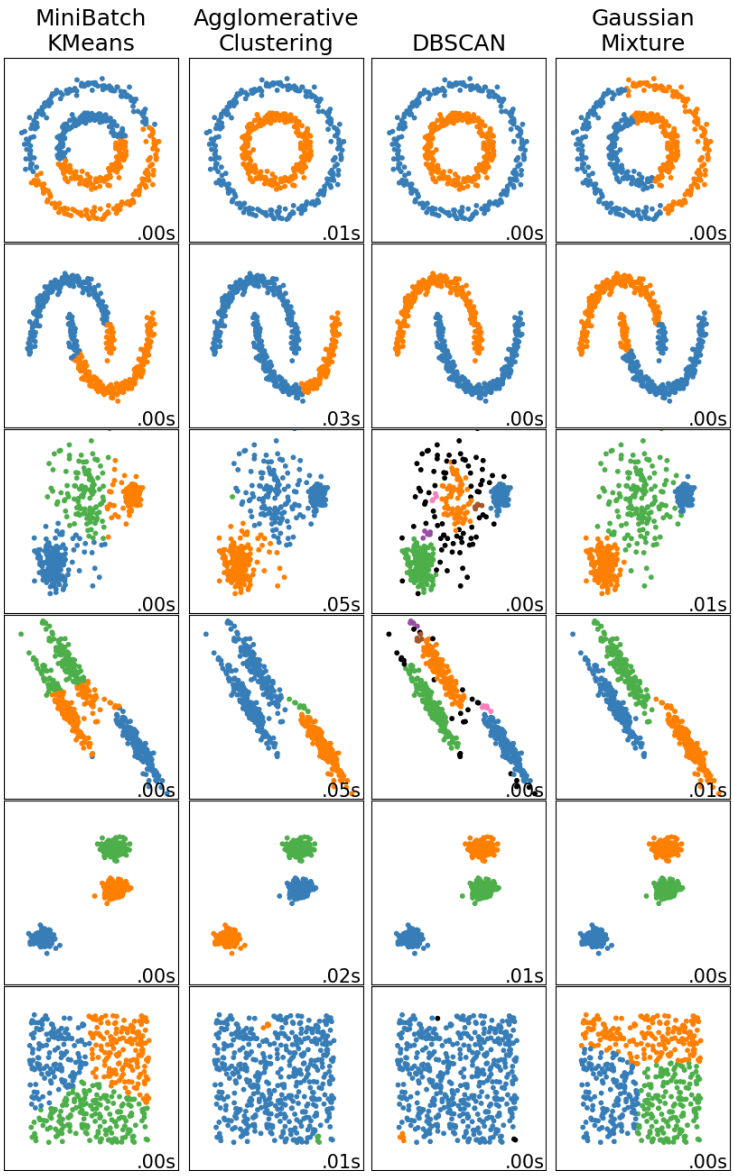


Рисунок – Сравнение методов кластеризации [3]

Список литературы

1. Chapter 10.3 Clustering Methods // An Introduction to Statistical Learning by Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshir. URL: https://www.statlearning.com/
2. Chapter 14.3. Cluster Analysis // The Elements of Statistical Learning by Trevor Hastie, Robert Tibshirani, Jerome Friedman. pp. 501–527. URL: https://web.stanford.edu/~hastie/ElemStatLearn/
3. Clustering // Sklearn: User Guide. URL: https://scikit-learn.org/stable/modules/clustering.html
4. Christopher M. Bishop. 2006. Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag, Berlin, Heidelberg.