Санкт-Петербургский государственный университет

Прикладная математика, программирование и искусственный интеллект

Отчет по учебной практике (научно-исследовательской работе)

Классические методы кластеризации

Выполнил:

Иващенко Сергей Юрьевич

Кафедра прикладной кибернетики

Санкт-Петербург

2023

Содержание

Введение

Кластер-анализ – способ группировки многомерных объектов, основанный на представлении результатов отдельных наблюдений точками подходящего геометрического пространства с последующим выделением групп, как «сгустков» этих точек (кластеров). Иными словами, кластеризация – классификация объектов на основе их сходства друг с другом, при которой принадлежность этих объектов каким-либо классам не задаётся.

Кластерный анализ как научное направление зародился в середине 60-х годов и с тех пор бурно развивается.

В современной жизни кластеризация применяется почти во всех сферах, где имеет место анализ и обработка данных: искусственный интеллект и машинное обучение, распознавание образов и обработка изображений, работа с документами и квантование. Также известны случаи её применения в биологии, медицине и даже археологии.

Быстрое развитие кластерного анализа определило разнообразие различных техник и методологий кластеризации. Основная задача – разбить выборку объектов на непересекающиеся подмножества, чьи объекты внутри похожи между собой по заданной метрике, но отличаются от объектов другого подмножества. Однако сама формулировка даёт понять, что задача кластеризации не совсем корректно поставлена. Причины неоднозначности: отсутствие критерия качества кластеризации, отсутствие заранее заданного числа кластеров, субъективные меры схожести между объектами.

Данные причины приводят к следующему выводу: методы кластеризации могут принципиально отличаться и давать разный результат.

Цель данной работы – исследовать классические методы кластеризации.

В работе будут изложены описания и особенности следующих алгоритмов: иерархический метод, алгоритм теории графов, EM-кластеризация, k-means, FOREL. Также предоставлено сравнение результатов алгоритмов на наборе данных.

Постановка задачи

Цель данной работы – исследовать классические методы кластеризации.

Чтобы достичь этой цели необходимо решить следующие задачи:

1. Изучить и выявить особенности следующих алгоритмов: иерархический метод, алгоритм теории графов, EM-кластеризация, k-means, FOREL
2. Сравнить результаты работы алгоритмов на наборе данных

Глава 1. Методы кластеризации

1.1 Иерархический метод

Иерархические алгоритмы кластеризации, называемые также алгоритмами таксономии, строят не одно разбиение выборки на непересекающиеся классы, а систему вложенных разбиений. Результат таксономии обычно представляется в виде таксономического дерева — дендрограммы.

Среди алгоритмов иерархической кластеризации различаются два основных типа: дивизимные и агломеративные. Дивизимные алгоритмы изначально собирают все элементы выборки в один кластер, а затем разбивают его на более мелкие. В агломеративных алгоритмах каждый объект представляет собой одноточечный кластер, а затем объединяется в более крупный.

Сначала каждый элемент является кластером. Для одноэлементных кластеров определяется функция расстояния:

$$R(\{x\},\{x\prime\}) = \rho(x,x\prime)$$

Затем запускается процесс слияний. На каждой итерации вместо пары самых близких кластеров U и V образуется новый кластер W = U∪V. Расстояние от нового кластера W до любого другого кластера S вычисляется по расстояниям R(U, V), R(U, S) и R(V, S), которые к этому моменту уже должны быть известны:

$$R(U \cup V, S) = \alpha_U R(U,S) + \alpha_V R(V,S) + \beta R(U,V) + \gamma |R(U,S) - R(V,S)|$$

α_U, α_V, β, γ — числовые параметры.

На практике применяются следующие расстояния:

Расстояние ближнего соседа:

$$R^{\text{б}}(W,S) = \min_{w\in W, s\in S} \rho(w,s); \qquad \alpha_U = \alpha_V = \tfrac{1}{2},\ \beta = 0,\ \gamma = -\tfrac{1}{2}.$$

Расстояние дальнего соседа:

$$R^{\text{д}}(W,S) = \max_{w\in W, s\in S} \rho(w,s); \qquad \alpha_U = \alpha_V = \tfrac{1}{2},\ \beta = 0,\ \gamma = \tfrac{1}{2}.$$

Среднее расстояние:

$$R^{\text{с}}(W,S) = \frac{1}{|W||S|} \sum_{w\in W} \sum_{s\in S} \rho(w,s); \qquad \alpha_U = \frac{|U|}{|W|},\ \alpha_V = \frac{|V|}{|W|},\ \beta = \gamma = 0.$$

Расстояние между центрами:

$$R^{\text{ц}}(W,S) = \rho^2\Big(\sum_{w\in W} \frac{w}{|W|}, \sum_{s\in S} \frac{s}{|S|}\Big); \qquad \alpha_U = \frac{|U|}{|W|},\ \alpha_V = \frac{|V|}{|W|},\ \beta = -\alpha_U\alpha_V,\ \gamma = 0.$$

Расстояние Уорда:

$$R^{\text{у}}(W,S) = \frac{|S||W|}{|S|+|W|}\rho^2\Big(\sum_{w\in W} \frac{w}{|W|}, \sum_{s\in S} \frac{s}{|S|}\Big); \quad \alpha_U = \frac{|S|+|U|}{|S|+|W|},\ \alpha_V = \frac{|S|+|V|}{|S|+|W|},\ \beta = \frac{-|S|}{|S|+|W|},\ \gamma = 0.$$

С помощью псевдокода алгоритм можно представить следующим образом:

1: инициализировать множество кластеров C_1:
$t := 1; \quad C_t = \{\{x_1\}, \ldots, \{x_\ell\}\};$
2: **для всех** $t = 2, \ldots, \ell \quad (t$ — номер итерации):
3: найти в C_{t-1} два ближайших кластера:
$(U, V) := \arg\min_{U \neq V} R(U, V);$
$R_t := R(U, V);$
4: изъять кластеры U и V, добавить слитый кластер $W = U \cup V$:
$C_t := C_{t-1} \cup \{W\} \setminus \{U, V\};$
5: **для всех** $S \in C_t$
6: вычислить расстояние $R(W, S)$ по формуле Ланса-Уильямса;

Преимущества алгоритма:

1. Алгоритм предоставляет возможность строить дендрограммы, что является удобной визуализацией кластеризации и инструментом для определения количества кластеров
2. Особенно эффективен, когда набор данных содержит реальные иерархические отношения
3. Обилие модификаций, позволяющие увеличить эффективность (например, на основе редуктивности)

Недостатки алгоритма:

1. Высокая вычислительная стоимость $O(n^2)$
2. Пример чёткой кластеризации (один объект принадлежит одному кластеру)

1.2 Алгоритм теории графов (Алгоритм кратчайшего незамкнутого пути)

Суть таких алгоритмов заключается в том, что выборка объектов представляется в виде графа, вершинам которого соответствуют объекты, а ребра имеют вес, равный «расстоянию» между объектами.

Алгоритм кратчайшего незамкнутого пути строит граф из $\ell-1$ рёбер так, чтобы они соединяли все ℓ точек и обладали минимальной суммарной длиной. Такой граф называется кратчайшим незамкнутым путём (КНП), минимальным покрывающим деревом или каркасом. Далее удаляются $K-1$ самых длинных рёбер, и связный граф распадается на K кластеров.

Псевдокод алгоритма:

1: Найти пару точек (i, j) с наименьшим ρ_{ij} и соединить их ребром;
2: **пока** в выборке остаются изолированные точки
3: найти изолированную точку, ближайшую к некоторой неизолированной;
4: соединить эти две точки ребром;
5: удалить $K-1$ самых длинных рёбер;

Достоинства:

1. Информативность и наглядность
2. Простота реализации (интуитивно понятные графы)
3. Количество кластеров – входной параметр

Недостатки:

1. Наличие разреженного фона («шума») между кластерами приводит к уменьшению информативности результата кластеризации
2. Высокая трудоёмкость $O(\ell^3)$

1.3 EM-кластеризация

EM-алгоритм заключается в итерационном повторении двух шагов. На E-шаге по формуле Байеса вычисляются скрытые переменные g_{iy}. Значение g_{iy} равно вероятности того, что объект $x_i \in X^\ell$ принадлежит кластеру $y \in Y$. На M-шаге уточняются параметры каждого кластера (μ_y, Σ_y), при этом используются скрытые переменные g_{iy}.

Для пущего понимания алгоритма уместно привести следующие гипотезы.

Гипотеза 1.1 (о вероятностной природе данных). Объекты выборки $X\ell$ появляются случайно и независимо согласно вероятностному распределению, представляющему собой смесь распределений

$$p(x) = \sum_{y \in Y} w_y p_y(x), \qquad \sum_{y \in Y} w_y = 1,$$

где $p_y(x)$ — функция плотности распределения кластера y, w_y — неизвестная априорная вероятность появления объектов из кластера y.

Гипотеза 1.2 (о форме кластеров). Объекты описываются n числовыми признаками $f_1(x)$, ..., $f_n(x)$, $X = R^n$. Каждый кластер $y \in Y$ описывается n-мерной гауссовской плотностью $p_y(x) = N(x; \mu_y, \Sigma_y)$ с центром $\mu_y = (\mu_{y1}, ..., \mu_{yn})$ и диагональной ковариационной матрицей $\Sigma_y = \mathrm{diag}(\sigma^2_{y1}, ..., \sigma^2_{yn})$.

При этих гипотезах задача кластеризации совпадает с задачей разделения смеси вероятностных распределений, и для её решения можно применить EM-алгоритм.

Алгоритм, представленный псевдокодом:

1: начальное приближение для всех кластеров $y \in Y$:
$w_y := 1/|Y|$;
$\mu_y :=$ случайный объект выборки;
$\sigma^2_{yj} := \frac{1}{\ell|Y|} \sum_{i=1}^{\ell} (f_j(x_i) - \mu_{yj})^2, \ j = 1, \ldots, n$;
2: **повторять**
3: E-шаг (expectation):
$$g_{iy} := \frac{w_y p_y(x_i)}{\sum_{z \in Y} w_z p_z(x_i)}, \ y \in Y, \ i = 1, \ldots, \ell;$$
4: M-шаг (maximization):
$$w_y := \frac{1}{\ell} \sum_{i=1}^{\ell} g_{iy}, \ y \in Y;$$
$$\mu_{yj} := \frac{1}{\ell w_y} \sum_{i=1}^{\ell} g_{iy} f_j(x_i), \ y \in Y, \ j = 1, \ldots, n;$$
$$\sigma^2_{yj} := \frac{1}{\ell w_y} \sum_{i=1}^{\ell} g_{iy} (f_j(x_i) - \mu_{yj})^2, \ y \in Y, \ j = 1, \ldots, n;$$
5: Отнести объекты к кластерам по байесовскому решающему правилу:
$y_i := \arg\max_{y \in Y} g_{iy}, \ i = 1, \ldots, \ell$;
6: **пока** y_i не перестанут изменяться;

Преимущества алгоритма EM:

1. Наличие формальной статистической основы.
2. Линейное увеличение сложности при росте объема данных (масштабируемость).
3. Устойчивость к шумам и пропускам в данных.
4. Возможность построения желаемого числа кластеров.
5. Быстрая сходимость при удачной инициализации.
6. Нечёткая кластеризация

Недостатки алгоритма EM:

1. Предположение о нормальности всех переменных модели (измерений данных) является нереалистичным, что делает алгоритм эвристическим.
2. При неудачной инициализации сходимость алгоритма может оказаться медленной.
3. Алгоритм может остановиться в локальном минимуме и дать квазиоптимальное решение (приближенное к оптимальному, но выбираемое из ограниченного количества вариантов)

1.4 Алгоритм K-means

По своей сути k-means является упрощением EM-алгоритма. Главное отличие в том, что в EM-алгоритме каждый объект x_i распределяется по всем кластерам с вероятностями $g_{iy} = P\{y_i = y\}$. В алгоритме k-средних каждый объект жёстко приписывается только к одному кластеру.

Второе отличие в том, что в k-means форма кластеров не настраивается (в общем виде, но с модификациями такая возможность появляется).

Алгоритм k-means крайне чувствителен к выбору начальных приближений центров. Случайная инициализация центров на шаге 1 может приводить к плохим кластеризациям. Для формирования начального приближения лучше выделить k наиболее удалённых точек выборки: первые две точки выделяются по максимуму всех попарных расстояний; каждая следующая точка выбирается так, чтобы расстояние от неё до ближайшей уже выделенной было максимально.

Кластеризация может оказаться неадекватной и в том случае, если изначально будет неверно угадано число кластеров. Стандартная рекомендация — провести кластеризацию при различных значениях k и выбрать то, при котором достигается резкое улучшение качества кластеризации по заданному функционалу.

Кластеризация с помощью k-means:

1: сформировать начальное приближение центров всех кластеров $y \in Y$:
μ_y — наиболее удалённые друг от друга объекты выборки;
2: **повторять**
3: отнести каждый объект к ближайшему центру (аналог E-шага):
$y_i := \arg\min_{y \in Y} \rho(x_i, \mu_y), \ \ i = 1, \ldots, \ell;$
4: вычислить новое положение центров (аналог M-шага):
$\mu_{yj} := \dfrac{\sum_{i=1}^{\ell} [y_i = y] f_j(x_i)}{\sum_{i=1}^{\ell} [y_i = y]}, \ \ y \in Y, \ \ j = 1, \ldots, n;$
5: **пока** y_i не перестанут изменяться;

Преимущества:

1. Простота реализации
2. Масштабируемость до огромных наборов данных
3. Метод очень быстро обучается на новых примерах
4. Поддержка сложных форм и размеров (в модификациях)

Недостатки:

1. Чувствительность к выбросам
2. Трудоемкость выбора k
3. Уменьшение масштабируемости

1.5 Алгоритм FOREL

Пусть задана некоторая точка $x_0 \in X$ и параметр R. Выделяются все точки выборки $x^i \in X^\ell$, попадающие внутрь сферы $\rho(x_i, x_0) < R$, и точка x_0 переносится в центр тяжести выделенных точек. Эта процедура повторяется до тех пор, пока состав выделенных точек, а значит и положение центра, не перестанет меняться.

Доказано, что эта процедура сходится за конечное число шагов. При этом сфера перемещается в место локального сгущения точек. Центр сферы x_0 в общем случае не является объектом выборки, потому и называется формальным элементом.

Для вычисления центра необходимо, чтобы множество объектов X было не только метрическим, но и линейным векторным пространством. Это требование естественным образом выполняется, когда объекты описываются числовыми признаками. Однако существуют задачи, в которых изначально задана только метрика, а сложение и умножение на число не определены на X. Тогда в качестве центра сферы можно взять тот объект обучающей выборки, для которого среднее расстояние до других объектов кластера минимально.

При этом заметно увеличивается трудоёмкость алгоритма. Если в линейном пространстве для вычисления центра требуется O(к) операций, то в метрическом – $O(n^2)$, где n – число точек в кластере. Алгоритм можно несколько ускорить, если заметить, что пересчёт центра при добавлении или удалении отдельной точки кластера требует лишь O(n) операций, а в линейном пространстве – O(1).

Различные варианты алгоритма FOREL отличаются способами объединения сфер в кластеры, способами варьирования параметра R, способами выбора начального приближения для точек x_0.

Один из вариантов представлен ниже:

Инициализировать множество некластеризованных точек:
$U := X^\ell$;
пока в выборке есть некластеризованные точки, $U \neq \varnothing$:
  взять произвольную точку $x_0 \in U$ случайным образом;
  повторять
    образовать кластер — сферу с центром в x_0 и радиусом R:
    $K_0 := \{x_i \in U \mid \rho(x_i, x_0) \leqslant R\}$;
    поместить центр сферы в центр масс кластера:
    $x_0 := \frac{1}{|K_0|} \sum_{x_i \in K_0} x_i$;
  пока центр x_0 не стабилизируется;
  пометить все точки K_0 как кластеризованные:
  $U := U \setminus K_0$;
применить алгоритм КНП к множеству центров всех найденных кластеров;
каждый объект $x_i \in X^\ell$ приписать кластеру с ближайшим центром;

Преимущества алгоритма:

1. Точность минимизации функционала качества (при удачном подборе параметра R)
2. Наглядность визуализации кластеризации
3. Сходимость алгоритма
4. Возможность операций над центрами кластеров - они известны в процессе работы алгоритма
5. Возможность подсчета промежуточных функционалов качества, например, длины цепочки локальных сгущений
6. Возможность проверки гипотез схожести и компактности в процессе работы алгоритма

Недостатки алгоритма:

1. Относительно низкая производительность (решается введение функции пересчета поиска центра при добавлении 1 объекта внутрь сферы)
2. Плохая применимость алгоритма при плохой разделимости выборки на кластеры
3. Неустойчивость алгоритма (зависимость от выбора начального объекта)
4. Произвольное по количеству разбиение на кластеры
5. Необходимость априорных знаний о ширине (диаметре) кластеров

Глава 2. Сравнение результатов работы методов кластеризации

2.1 Сведения о реализации алгоритмов

Далее будут предложены результаты применения каждого из описанных выше методов кластеризации.

Для реализации был выбран язык Python, так как он предоставляет множество библиотек для визуализации данных, а также возможность работы с классами, крайне удобными для данных алгоритмов. Так, практичным оказалось создание классов «Кластер», «Точка», «Граф» и «Ребро» (как вспомогательный для «Графа»). Все алгоритмы реализованы в соответствии с псевдокодами, представленными в описании каждого метода. Немалую роль сыграли также дополнительные функции для работы с матрицами (нахождение минимального и максимального элементов) и вычисления расстояния (Уорда для межкластерного и Евклидового как основного).

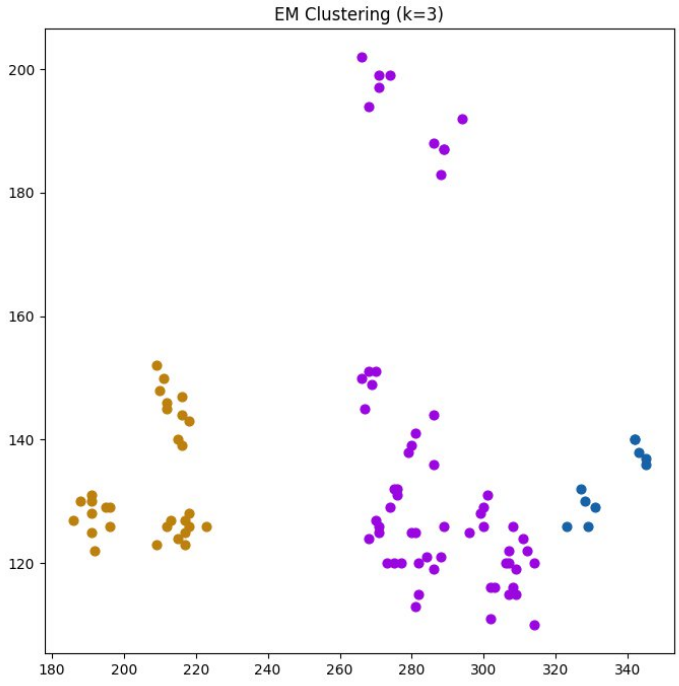
В качестве данных был выбран набор сведений о пациентах с сердечными заболеваниями, подготовленный Всемирной Организацией Здравоохранения. В качестве шкал (признаков элементов) были взяты 2 атрибута – холестерин в крови (ось x) и кровяное давление (ось y). Выбор обусловлен информативностью и показательностью данных. Дата-сет импортирован с сайта Kaggle.

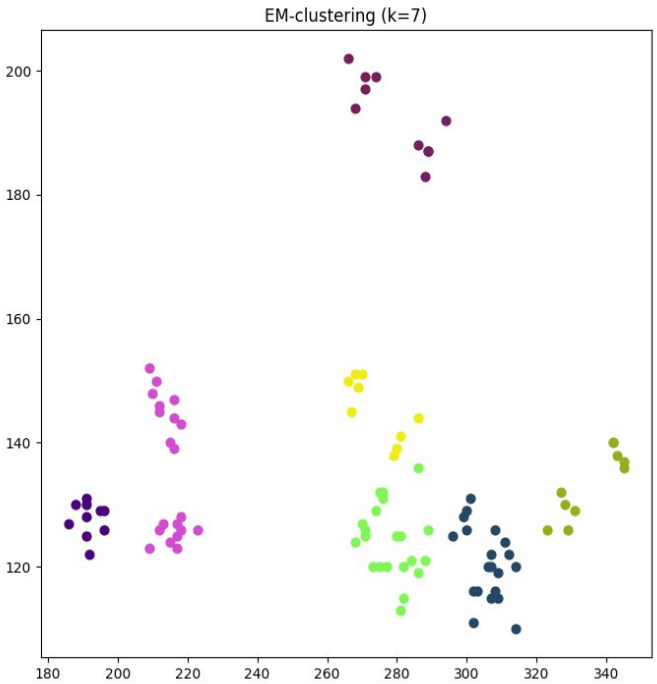
Чтобы получить полную картину, для каждого алгоритма приведены сразу два результата с разными параметрами. Для k-means и EM-алгоритма параметр – количество выходных кластеров (K). Для алгоритма теории графов – количество удалений самых длинных рёбер (K). Для иерархического метода – количество разделов (K). Для FOREL – радиус кластеров (R).

На графиках точки – объекты выборки. Элементы, принадлежащие разным кластерам отмечены разными цветами.

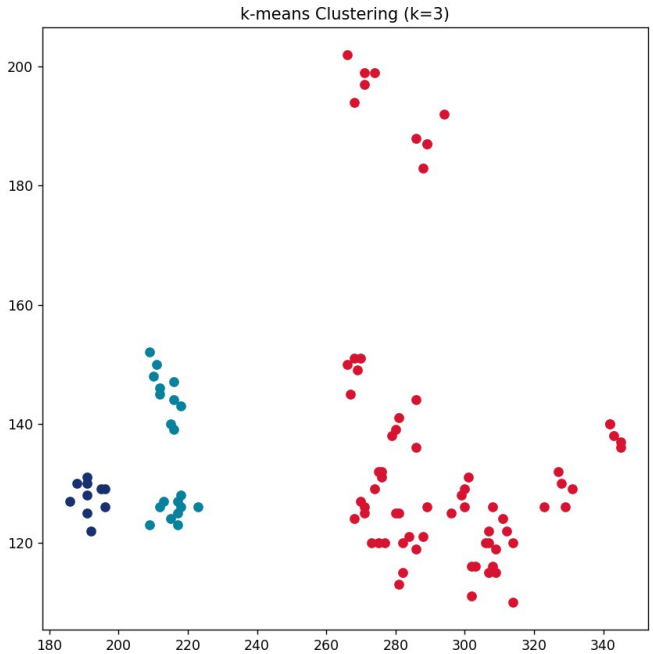
2.2 Результаты кластеризации

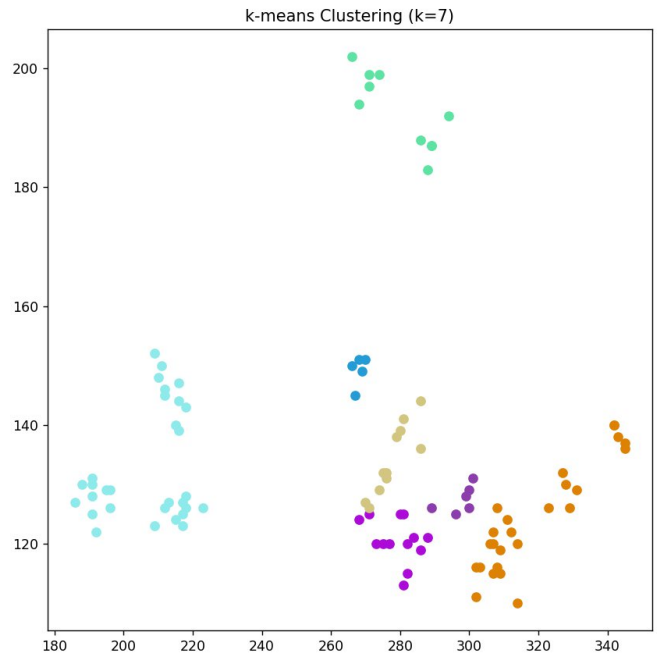
EM-алгоритм (k=3 и k=7)



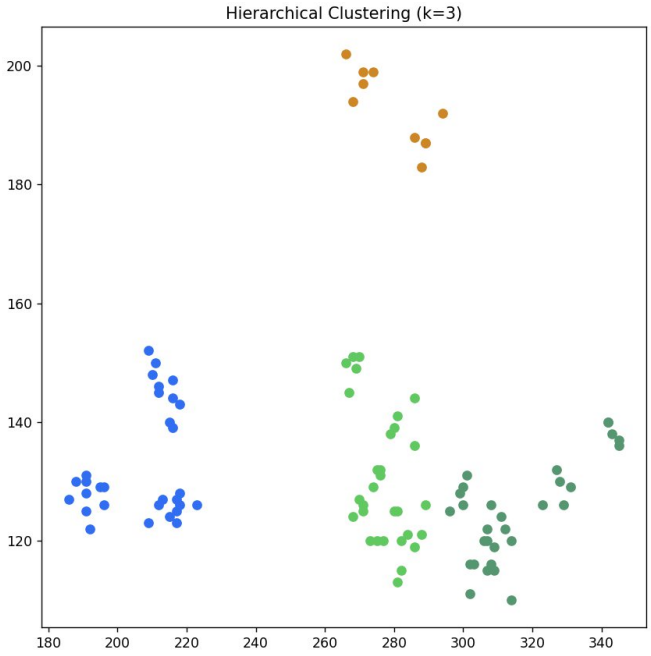


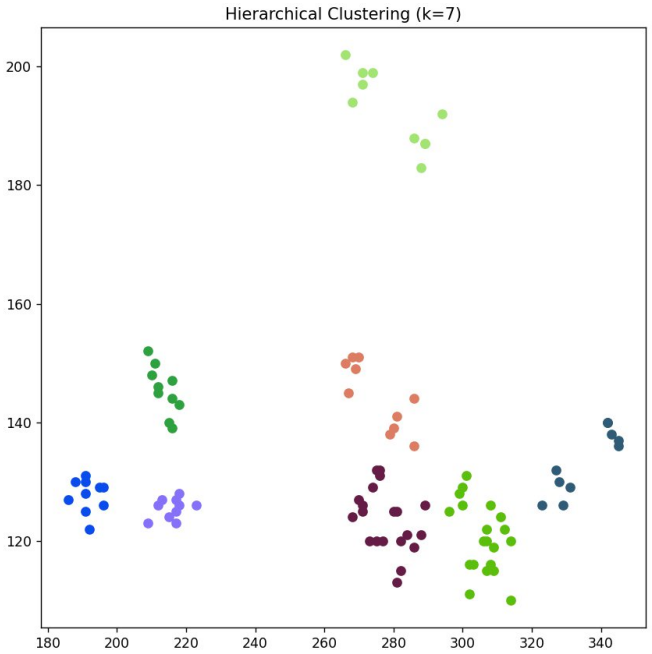
K-means (k=3 и k=7)



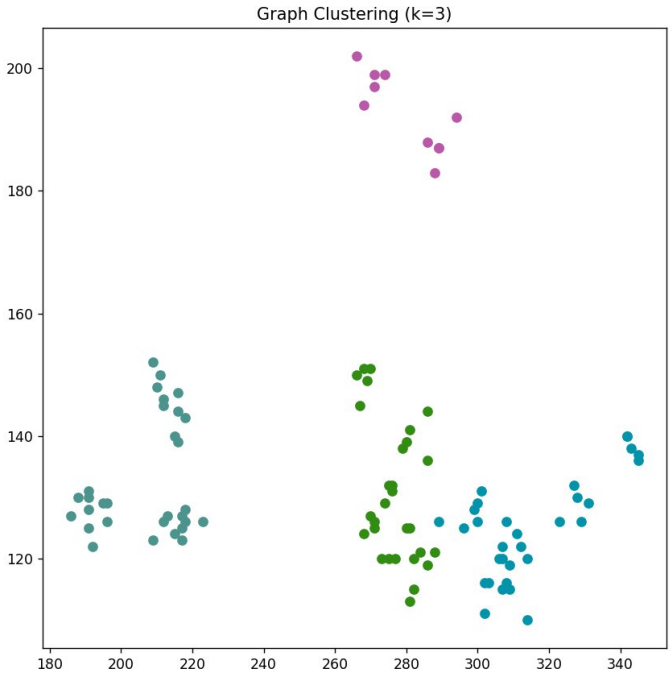


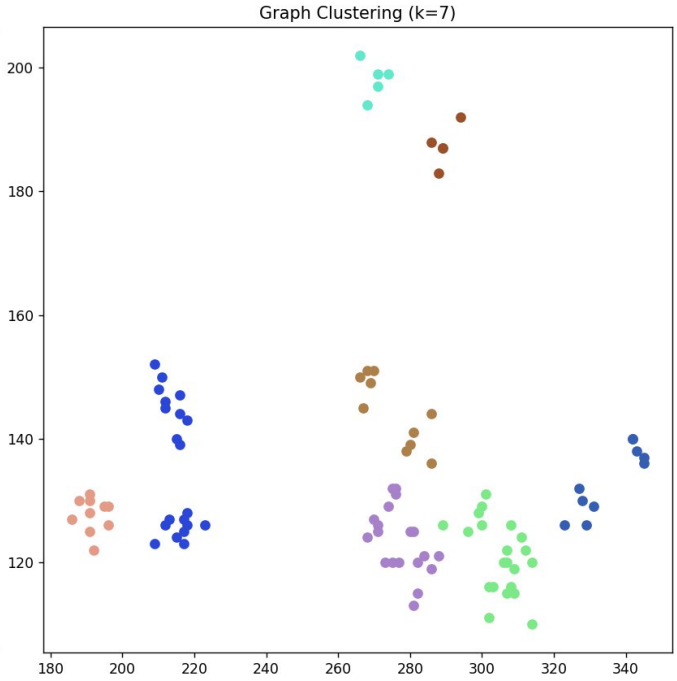
Иерархическая кластеризация (k=3 и k=7)



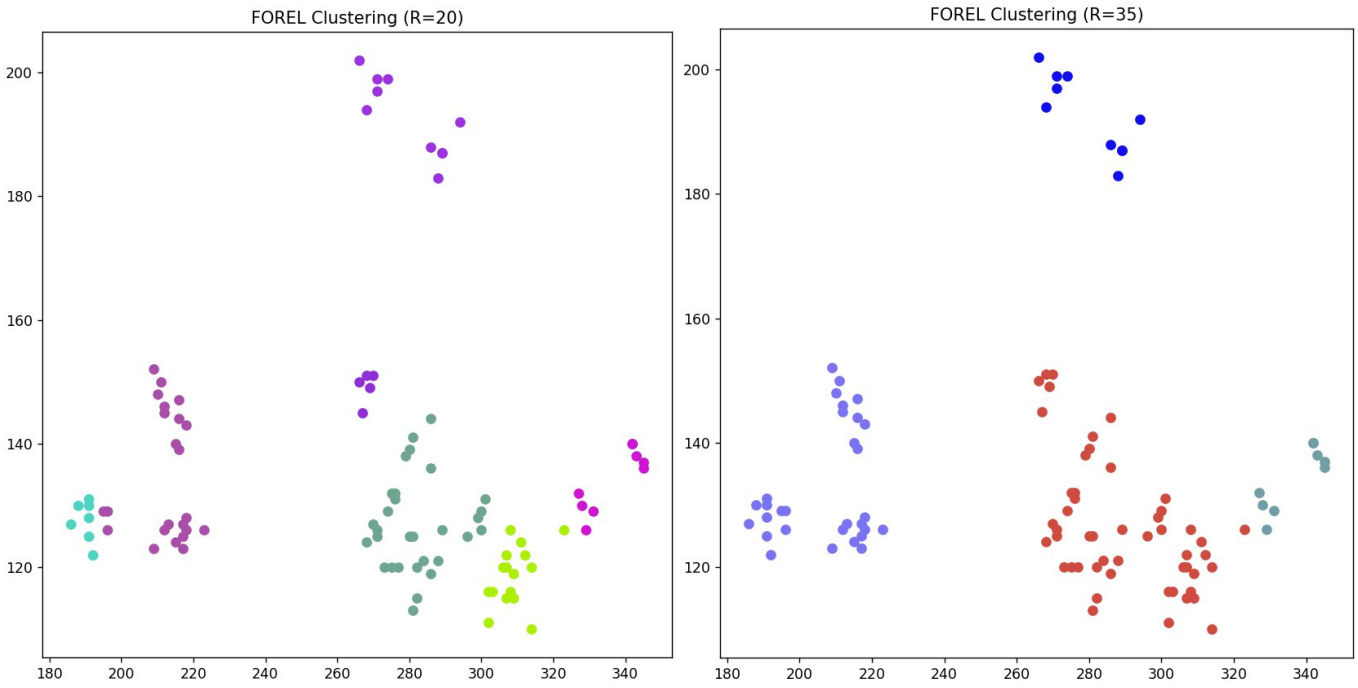


Алгоритм теории графов (k=3 и k=7)





Алгоритм FOREL (R=20 и R= 35)



2.3 Оценка результатов

Выше предоставлены результаты кластеризации. Как видно по графикам, наиболее эффективными с точки зрения информативности для малых параметров оказались иерархический метод, алгоритм FOREL, наименее – k-means. Для больших параметров следует пользоваться алгоритмами теории графов, иерархическим и EM-алгоритмом.

EM-кластеризация отличается хорошей точностью, а также устойчивостью к элементному «шуму» при достаточном количестве итераций (в исследовании порядка 100). В совокупности со свойствами нечёткости и масштабируемости является крайне эффективной, а строгая теоретическая база позволяет легко модифицировать алгоритм под практические нужды. Из недостатков – вычислительная стоимость.

K-means конкретно в этом исследовании показывает не самую высокую эффективность. Это связано с определённой ненастраиваемой формой кластеров, а также вероятностной природой выбора начальных значений. Однако из-за высокой эффективности и применимости к большим данным метод широко распространён.

Хороши алгоритмы теории графов и FOREL. Они предоставляют весьма наглядную кластеризацию, но не выдерживают проверку «шумом». Интуитивная простота реализации также выгодно выделяют эти методы.

Иерархическая кластеризация наиболее информативна. Помимо этого, она позволяет исследовать данные ещё глубже с помощью дендрограммы. С её же помощью задача кластеризации приобретает аналитические оттенки, а принятие решений (как одна из целей) становится намного удобней. В противовес – «тяжесть» алгоритма (но с возможностью уменьшения вычислительных расходов, как, например, с модификацией на основе редуктивности).

Заключение

В ходе этой работы были описаны основные методы кластеризации: иерархический метод, алгоритм теории графов, EM-кластеризация, k-means, FOREL. Помимо основных сведений и непосредственно алгоритмов, были упомянуты преимущества и недостатки техник.

Демонстрация результатов кластеризации даёт понять, что методы принципиально отличаются как алгоритмически, так и в своём результате. Некоторые методы удобнее на малом количестве данных, некоторые хорошо себя проявляют на больших, а некоторые имеют такие особенности, которые покрывают недостатки в виде сложности и вычислительных затрат.

В заключение следует отметить, что кластерный анализ – крайне интересное и перспективное направление. Кластеризация сама по себе является мощным инструментом для анализа и обработки информации. Она эффективна как для маленьких наборов данных, так и для больших дата-сетов, а применение находит в большинстве современных отраслей. Отсутствие универсальности и критерия качества кластеризации, неоднородность результатов, сложности с извлечением признаков, а также трудности с интеграцией предметной области в процесс представляют собой основные проблемы задачи кластеризации и зачастую именно с ними, а не с проблемой вычислительной стоимости приходится бороться исследователю.

Список литературы

1. Воронцов, К. В. Лекции по алгоритмам кластеризации и многомерного шкалирования / К. В. Воронцов. —М.: МГУ, 2007.
2. Jain A., Murty M., Flynn P. Data Clustering: A Review. // ACM Computing Surveys. 1999. Vol. 31, no. 3.
3. Мандель И. Д. Кластерный анализ. — М.: Финансы и Статистика, 1988.