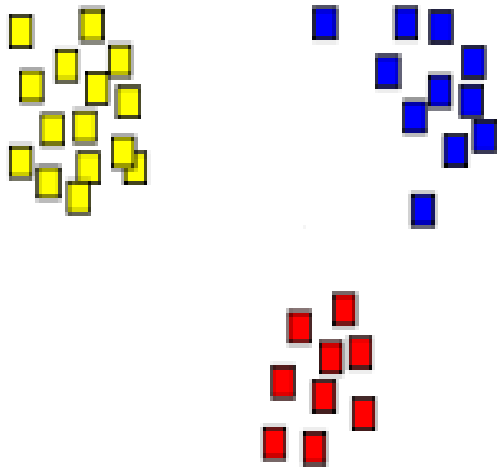
Кластеризация

Лекция 4

Лектор: Шевляков Артём

Определение

Кластеризация (clustering).
Дано множество объектов. Их нужно разбить на несколько групп (кластеров), состоящих из похожих друг на друга объектов.



Для чего нужна кластеризация?

1. Для вычисления степени сходства объектов. Например: содержание каких веб-страниц близко друг к другу, какие пользователи соцсети близки друг к другу по интересам…
2. Упростить дальнейшую обработку данных, разбить множество M на группы схожих объектов чтобы работать с каждой группой в отдельности.
3. Сократить объём хранимых данных, оставив по одному представителю (эталону) от каждого кластера (задачи сжатия данных).
4. Поиск выбросов (об этом говорилось на прошлой лекции).
5. Разбить признаки на кластеры и оставить по одному признаку из каждого кластера (отбор признаков).

Алгоритмы кластеризации делятся на группы

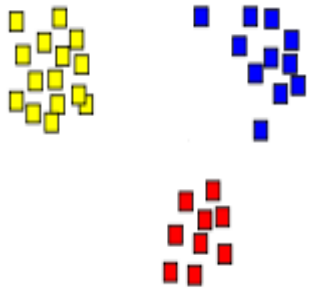
1. Алгоритмы, разбивающие данные на заданное число кластеров (то есть число кластеров – это входной параметр алгоритма). Пример: алгоритм k-means

2. Алгоритмы, в которых число кластеров не определено заранее, а вычисляется самим алгоритмом. Пример: алгоритм FOREL

Недостатки кластеризации каждого типа

1(тип).Человек может не угадать «нужное» число кластеров. Например, для объектов на картинке человек может запустить разбиение на 2 или 4 кластера.

2(тип). Алгоритм может выдать слишком много (мало) кластеров. Такая кластеризация бесполезна. Например, объекты на картинке могут быть разбиты на 1 или 10 кластеров (и это плохо).



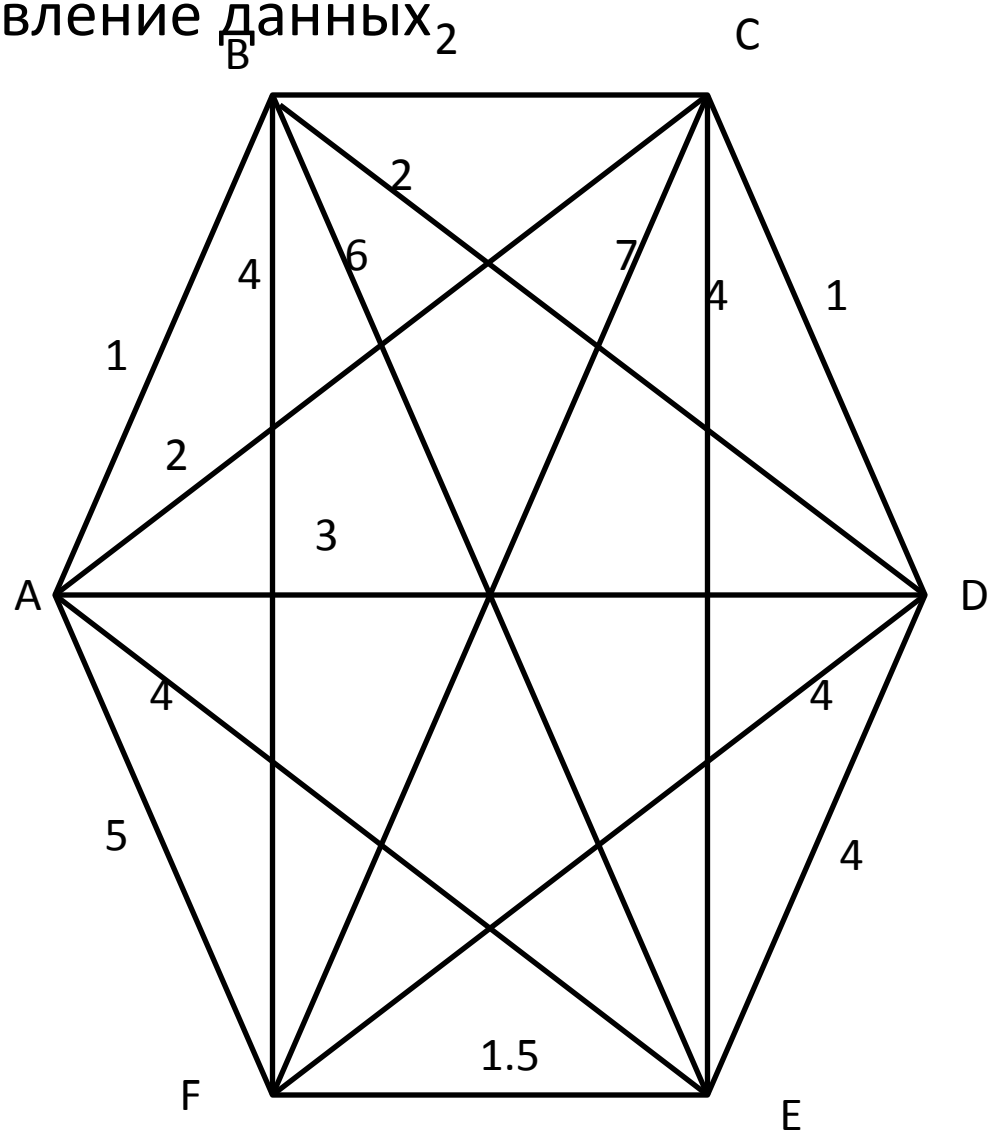
Не забывайте, что

Если алгоритм кластеризации использует метрику на множестве объектов, то значения всех признаков необходимо предварительно нормализовать.

Кластеризация с помощью графов (будут рассмотрены 2 алгоритма, принадлежащие двум типам алгоритмов кластеризации)

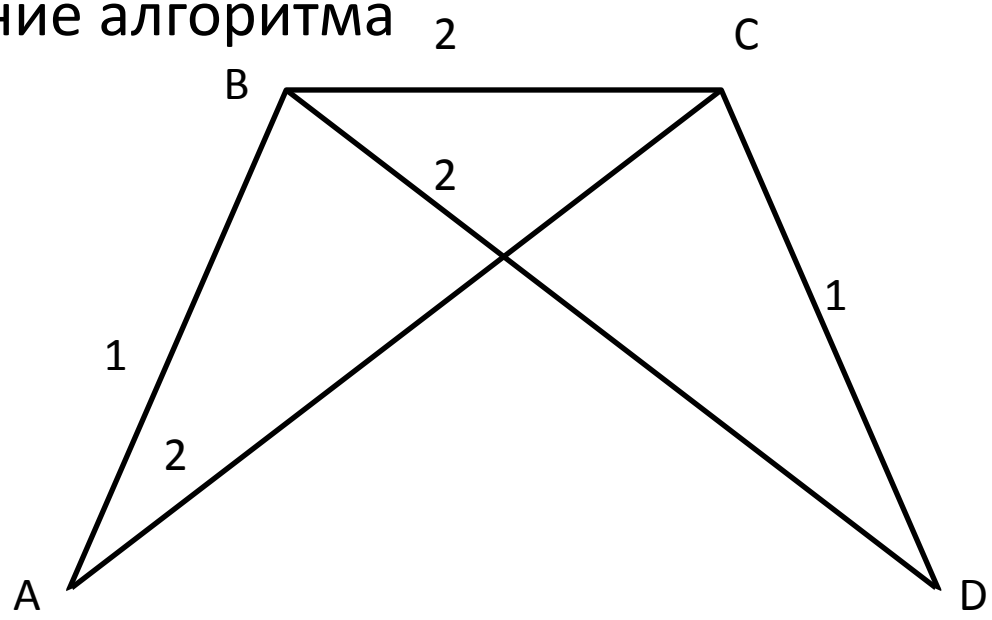
Представление данных 2

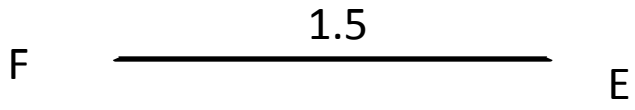
Необходимо вычислить расстояние между всеми парами объектов. Представить эти данные в виде графа (см. картинку)



Описание алгоритма

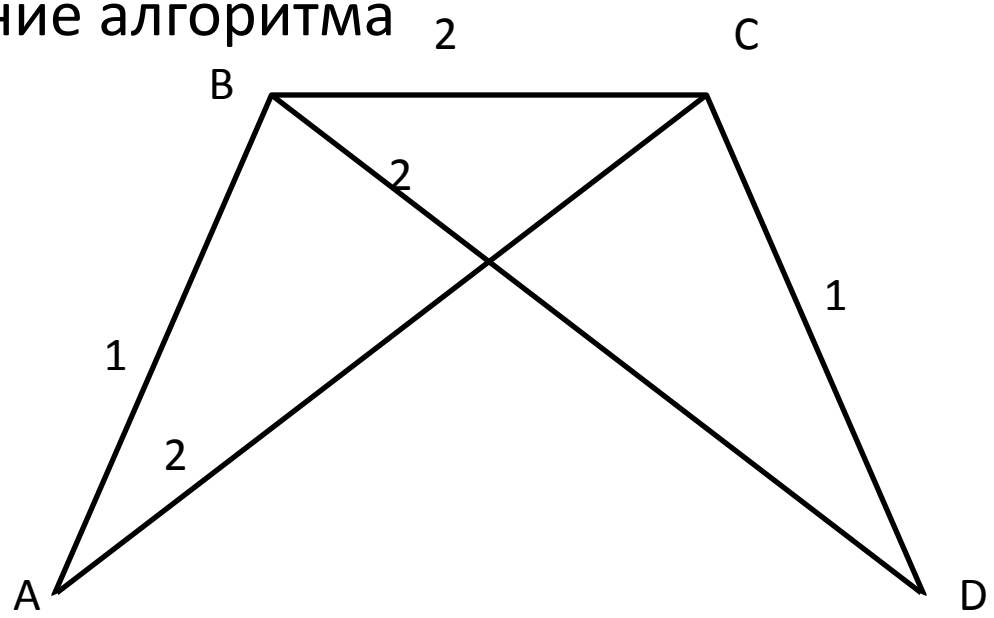
На вход алгоритма подается число *R.*
Удаляем все ребра в графе, метки которых *>R.*
Например, для *R=2* имеем картинку.
Кластеры – это…

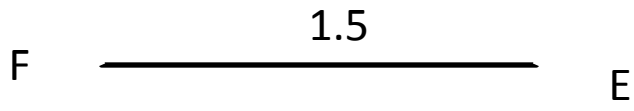




Описание алгоритма

На вход алгоритма подается число *R.*
Удаляем все ребра в графе, метки которых *>R.*
Например, для *R=2* имеем картинку.
Кластеры – это связные компоненты графа {A,B,C,D} и {E,F}

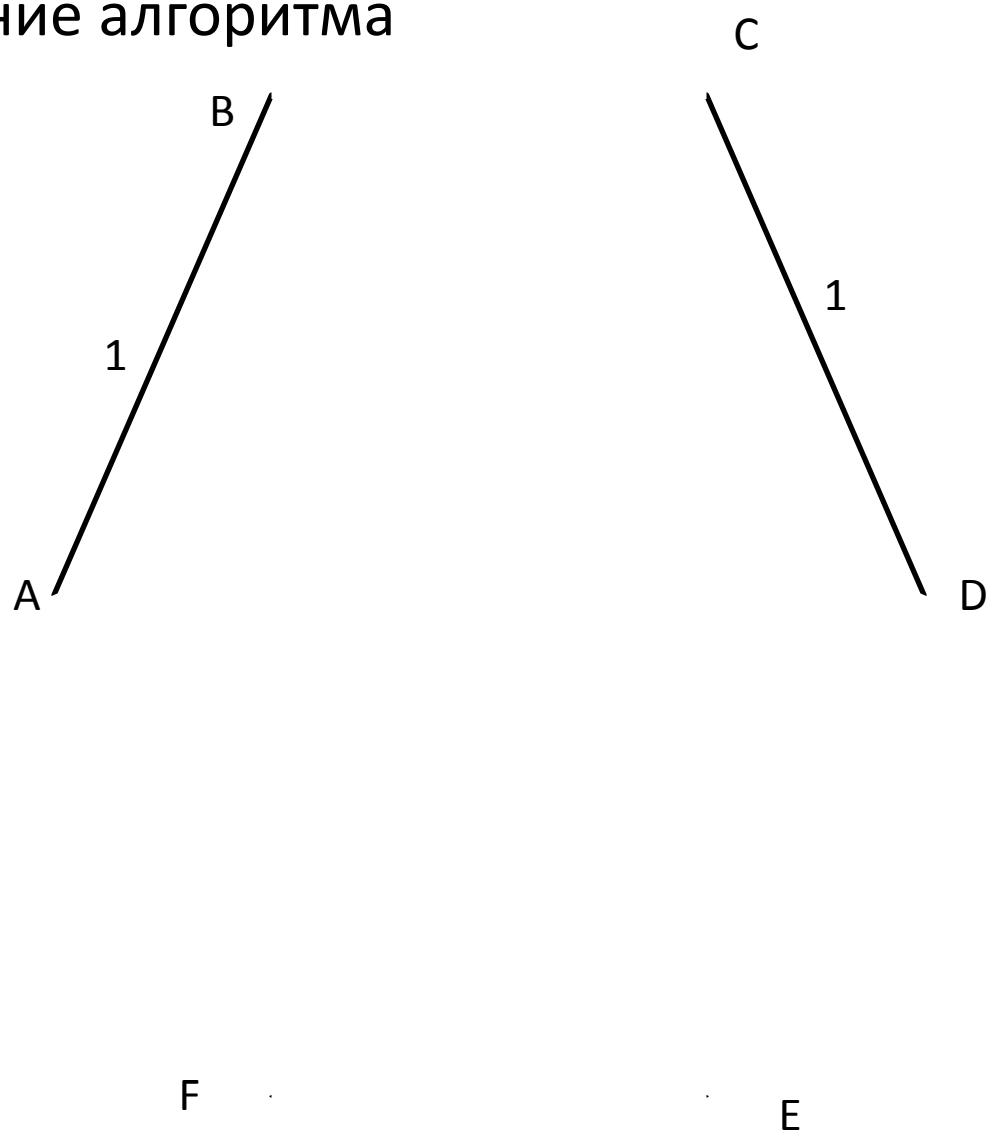




Описание алгоритма

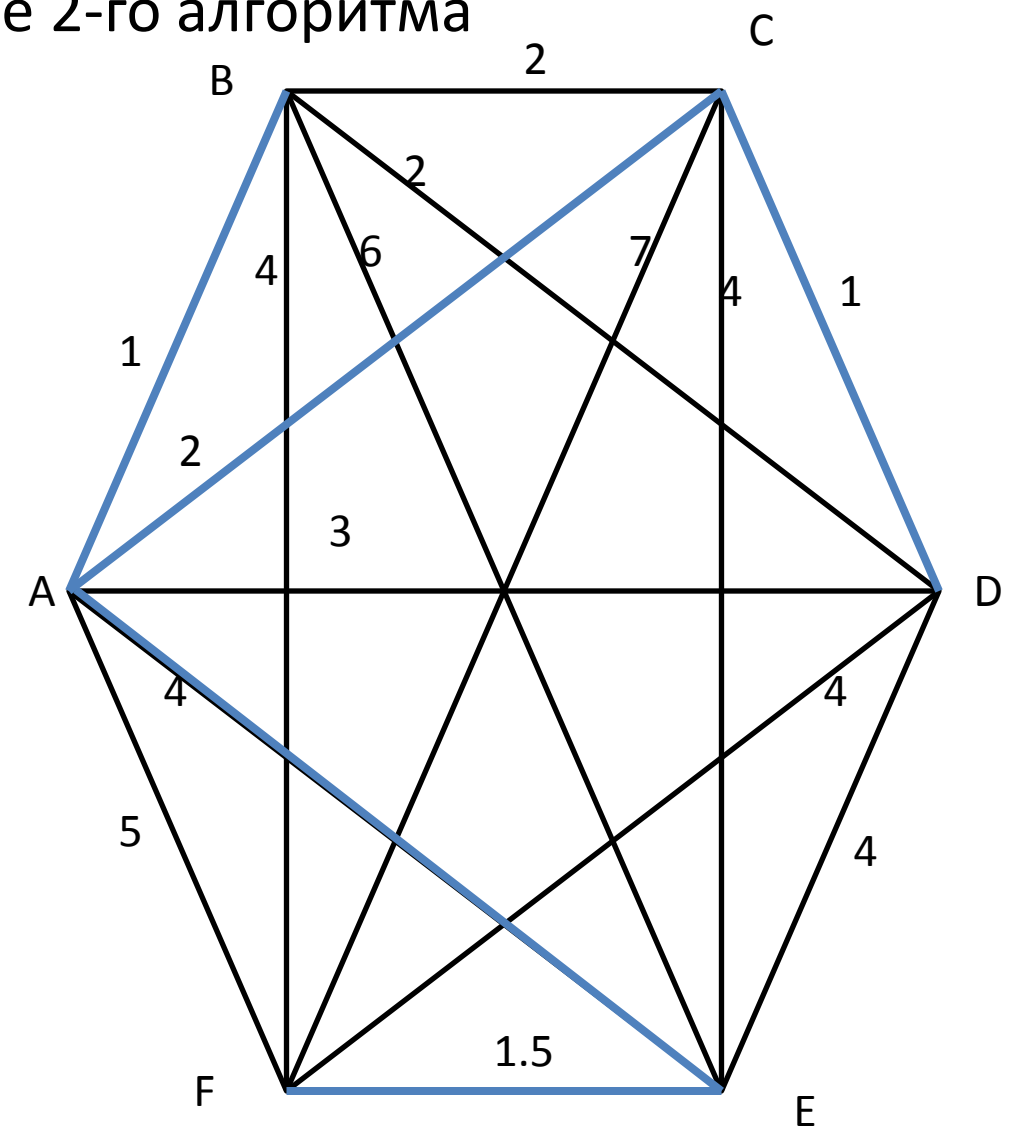
Если на вход алгоритма подать число 1.4, то получим 4 кластера {A,B},{C,D},{E},{F}.

Как видно, данный алгоритм не позволяет разбивать данные на фиксированное число кластеров.



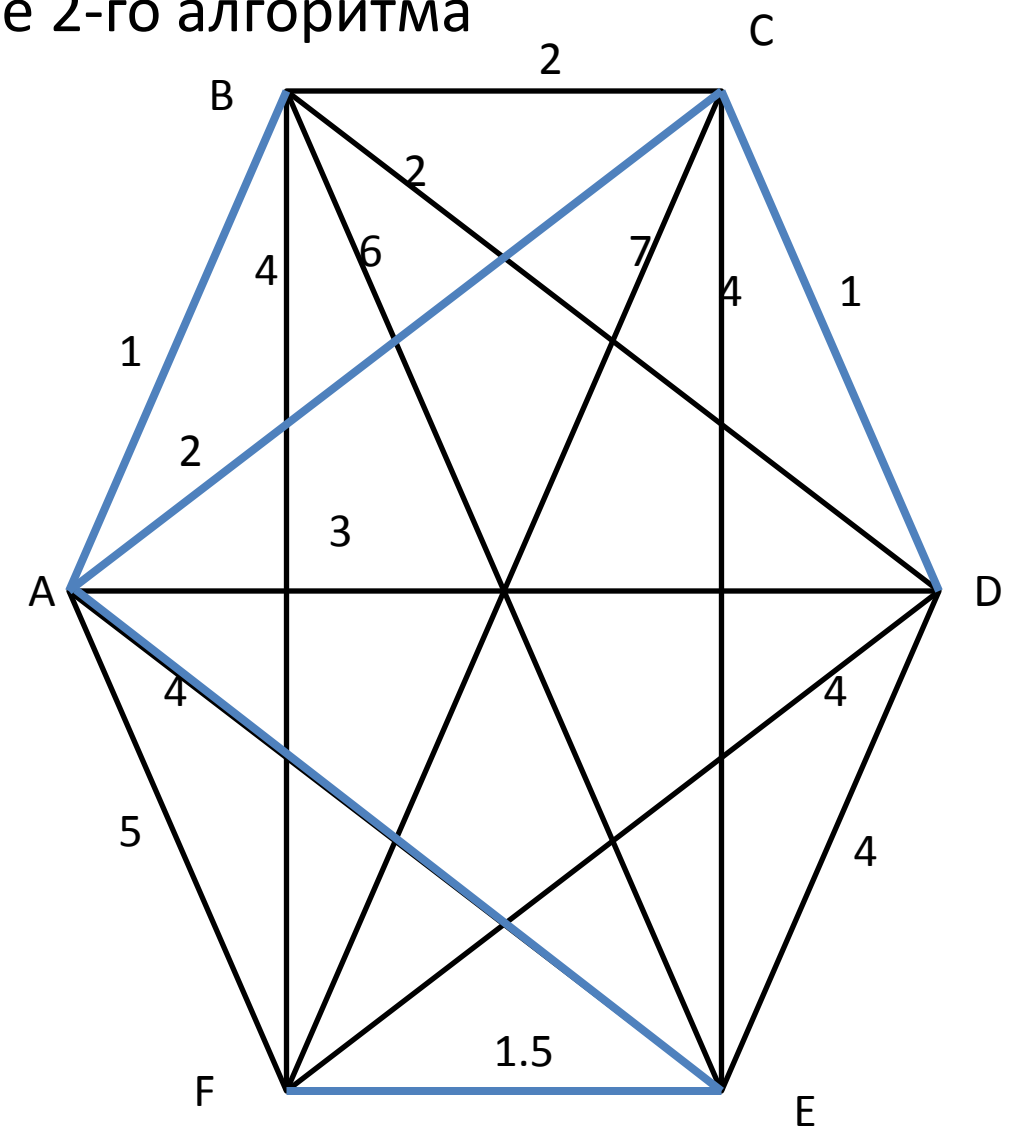
Описание 2-го алгоритма

На вход алгоритма подается число кластеров *k.*
1. Строим остовное дерево (это подграф, содержащий все вершины исходного графа и не имеющий циклов) минимальной длины.



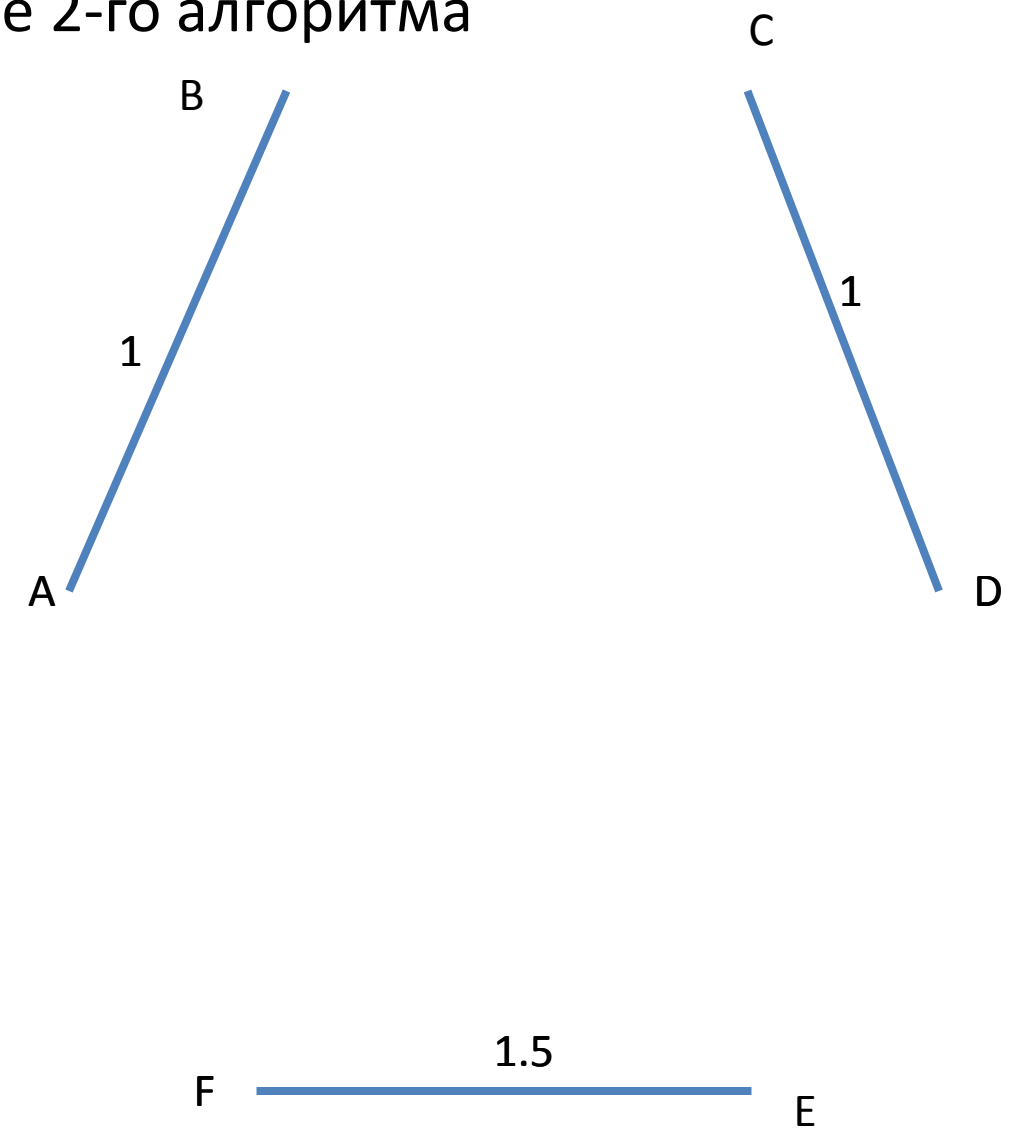
Описание 2-го алгоритма

2. Удаляем из дерева k-1 самых длинных ребер.
Например, для k=3 нужно удалить ребра AE и AC.



Описание 2-го алгоритма

2. Удаляем из дерева k-1 самых длинных ребер.
Например, для k=3 нужно удалить ребра AE и AC.
3. В один кластер попадают вершины из связных компонент.



Алгоритм FOREL (формальный элемент)

Главное свойство алгоритма: количество кластеров не определено заранее.

Идея: найти точки сгущения объектов, и эти сгущения объявить кластерами.

Описание алгоритма FOREL

Вход: число R.

Представление данных: объекты представляются точками в пространстве R^m

Шаг 1: В произвольную точку пространства добавляем новый формальный объект *F* (отсюда и название алгоритма).

Шаг 2: Пусть *K* – все объекты, до которых расстояние от *F* меньше *R*.

Шаг 3: находим центр тяжести (что это - см. ниже) объектов из множества *K*. Переносим туда объект *F*. Переходим на шаг 2.

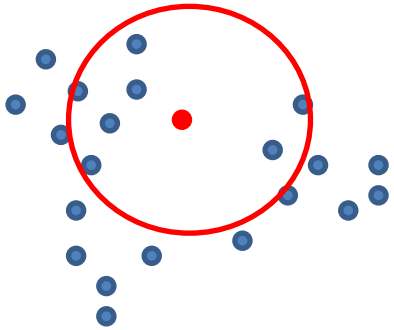
Нужно крутиться в цикле 2-3 до тех пор, пока множество *K* не стабилизируется.

Описание алгоритма FOREL

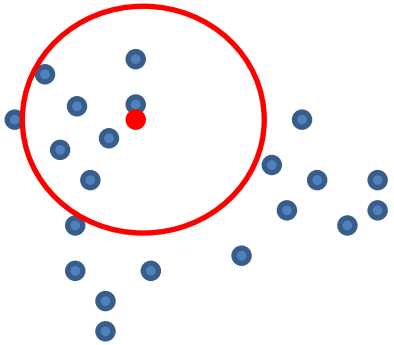
Шаг 4: Когда множество *K* стабилизируется, оно объявляется новым кластером. Объекты, попавшие в *K*, из выборки удаляются.

Шаг 5: Возвращаемся на шаг 1 если выборка не пуста, иначе конец работы.

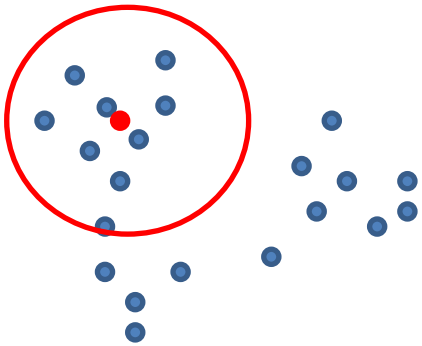
1



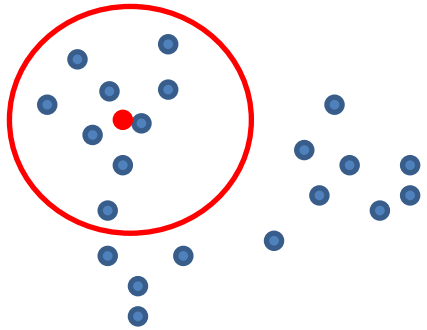
2



3



4



Новые точки в шар не попадают.
Точки внутри шара объявляются одним кластером.
Этот кластер исключается из выборки.
Процесс продолжается для оставшихся точек.

Как найти центр тяжести точек?

Да запросто!

Например, если объекты имеют 2 признака P,Q то центр тяжести будет обладать значениями признаков ($\bar{p}$,$\bar{q}$), где $\bar{p}$,$\bar{q}$ - среднее значение признаков.

Как найти центр тяжести точек?

Например, центр тяжести объектов

Объекты	P	Q
A	1	4
B	2	1
C	0	1

будет

	P	Q
Центр кластера	1	2

Алгоритм k-means (k-средних)

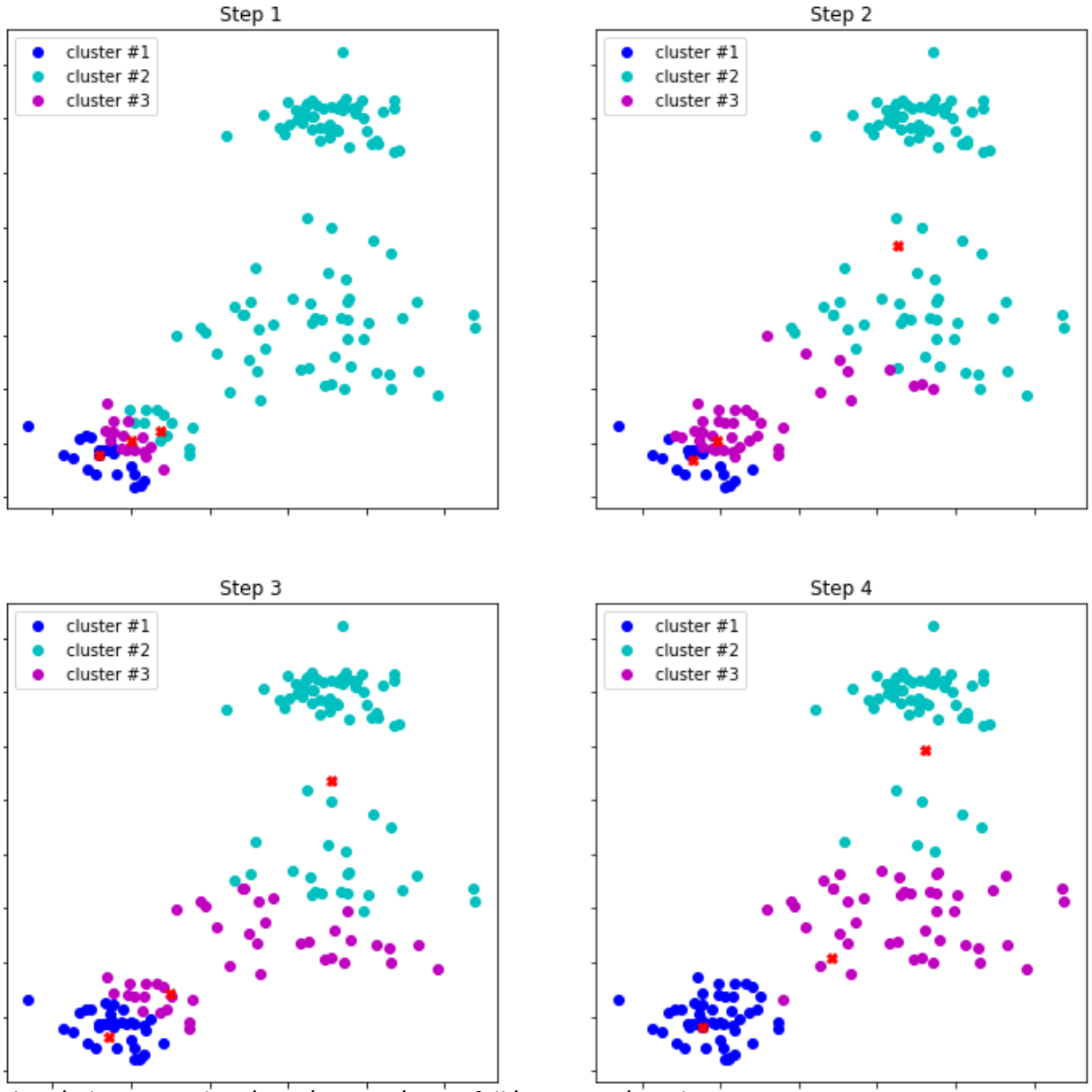
Главное свойство алгоритма: количество кластеров *k* определено заранее.

Идея реализации: одновременно происходит поиск всех центров кластеров.

Описание алгоритма k-means (одна из реализаций)

Вход: число кластеров k.
Представление данных: объекты представляются точками в пространстве R^m
Шаг 1: Генерируем k случайных точек – центры кластеров.
Шаг 2: Объект будет отнесен к тому кластеру, чей центр расположен ближе всех к этому объекту.
Шаг 3: Пересчитываются центры кластеров, возврат на Шаг 2.
Цикл 2-3 крутится, пока изменяются центры кластеров.

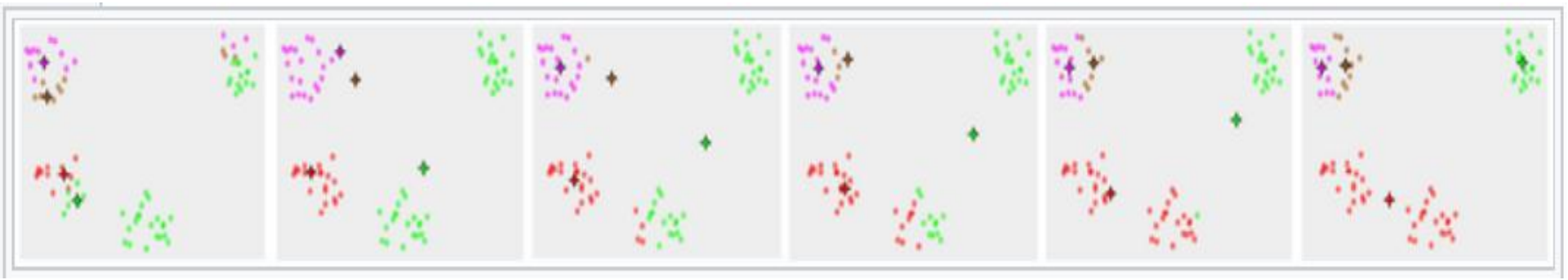
Пример работы алгоритма k-means



https://github.com/7bits/ml-course-7bits/tree/master/2017-fall/practises/04-clustering

Недостатки алгоритма k-means

Результат зависит от выбора исходных центров кластеров, их оптимальный выбор неизвестен.



https://ru.wikipedia.org/wiki/Метод_k-средних

Выбор оптимального числа кластеров

Выбор оптимального числа кластеров

Эта проблема актуальна для алгоритмов, в которых «число кластеров» является входным параметром. В частности, это актуально для k-means.

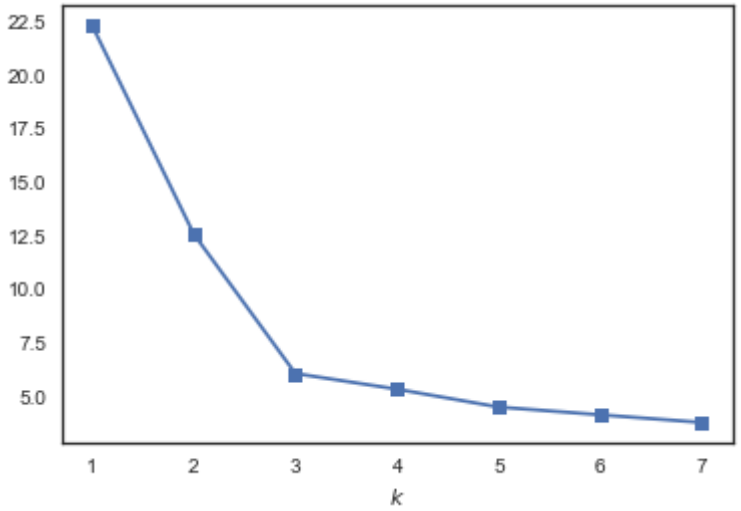
Идея: будем перебирать значения *k=1,2…* пока «качество кластеризации» не стабилизируется.

А что понимать под «качеством кластеризации»?
Пусть S_k – сумма расстояний от объектов до центров их кластеров (при условии, что объекты разбиты на *k* кластеров).
Тогда величину $|S_{k+1} - S_k|$ можно рассматривать как увеличение качества кластеризации при переходе от *k* кластеров к *(k+1)* кластеру.

Выбор оптимального числа кластеров

Таким образом, «качество кластеризации» стабилизируется для такого k, где величина $|S_{k+1} - S_k|$ становится небольшой.

На следующем графике по верт.оси отложено значение S_k. Для этого графика оптимальное значение $k=3$.



https://habrahabr.ru/company/ods/blog/325654/

Почему такое сложное правило для выбора k?

Попробуем разобраться:
Чему равно S_k при $k=n$ (число всех объектов)?

Напомню:

S_k = сумма расстояний от объектов до центров их кластеров.

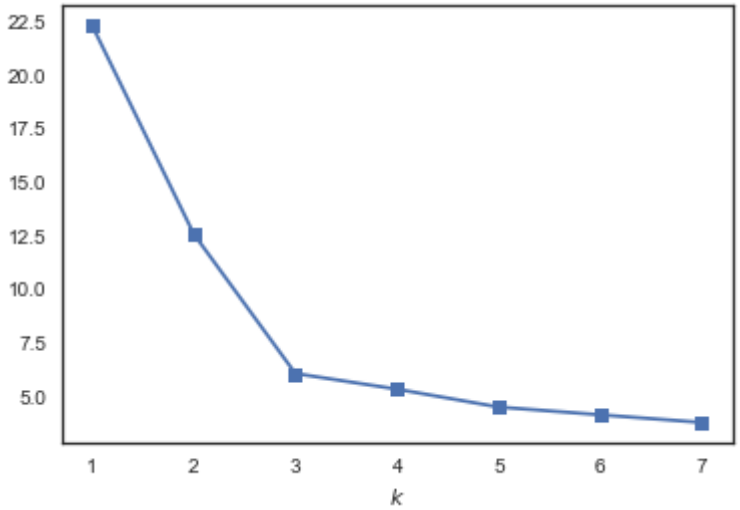
Почему такое сложное правило для выбора k?

Попробуем разобраться:
Чему равно S_k при $k=n$ (число всех объектов)?

Напомню:

S_k = сумма расстояний от объектов до центров их кластеров.

Не знаете ответа? Посмотрите, к какому значению стремиться график…



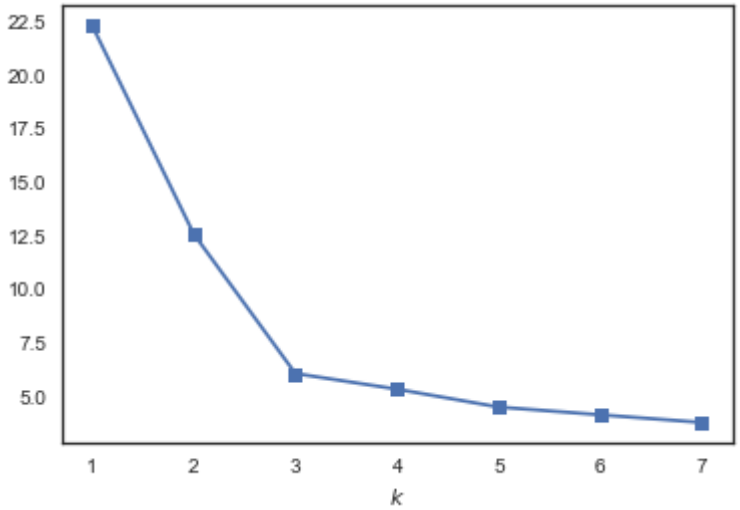
https://habrahabr.ru/company/ods/blog/325654/

Почему такое сложное правило для выбора k?

Действительно, $S_n = 0$ (n число всех объектов)!!!

То есть искать число кластеров k с минимальным S_k бессмысленно!

Вот и приходится исхитряться: искать такое k что S_k уменьшается не так сильно (когда дальнейшее увеличение числа кластеров уже не приводит к существенному улучшению качества).



https://habrahabr.ru/company/ods/blog/325654/

Кластеризация по столбцам

Кластеризация по столбцам

Дана таблица. Ее можно перевернуть (транспонировать)

Студент	Пол	Рост	Вес	Место на олимпиаде
Вася	1	172	107	3
Петя	1	185	64	4
Маша	0	168	61	2
Даша	0	201	85	1

А потом запускаем один из стандартных алгоритмов кластеризации!

	Вася	Петя	Маша	Даша
Пол	1	1	0	0
Рост	172	185	168	201
Вес	107	64	61	85
Место	3	4	2	1

Зачем это нужно делать?

Мы можем найти близкие (по значению) друг к другу признаки. Можно из каждого кластера оставить по одному признаку – и тем самым уменьшить размер данных.

Это иногда оправданно, так как огромное число признаков часто мешает анализу данных (поподробнее об этом в теме «Отбор признаков»)

Но есть и другое (неожиданное)приложение кластеризации по столбцам (см. след. слайды)

NMF
Non-negative matrix factorization
(развитие идеи о кластеризации столбцов и применение ее в рекомендательных системах)

Идея!

А если мы сможем найти новые признаки (выразив их через старые признаки), которые дают нетривиальную кластеризацию объектов?
Например, если для таблицы покупок найти 2 группы товаров, а потом разбить покупателей на 2 кластера – в зависимости от того, товары какой группы он предпочитает.

	Мука	Возд.шары	Пиво	Сахар	Чипсы
Покупатель1	0	3	8	0	1
Покупатель2	0	2	5	1	0
Покупатель3	5	0	1	10	0
Покупатель4	0	20	40	2	1
Покупатель5	10	0	1	10	1

В этом примере ответ простой:

Признаки нужно разбить на 2 группы: «товары для выпечки» и «товары для праздника».
Соответственно покупатели распадаются на кластеры:
1-й кластер: {1,2,4} – они покупают товары для праздника.
2-й кластер: {3,5} – они покупают товары для выпечки.

А как найти кластеры в общем случае?

	Мука	Возд.шары	Пиво	Сахар	Чипсы
Покупатель1	0	3	8	0	1
Покупатель2	0	2	5	1	0
Покупатель3	5	0	1	10	0
Покупатель4	0	20	40	2	1
Покупатель5	10	0	1	10	1

А для этого нужно знать, что такое матрицы

Спойлер: матрица – это таблица с числами. Например,

$\begin{pmatrix} 1 & 3 \\ 0 & 5 \end{pmatrix}, \begin{pmatrix} 2 & 3 \\ 1 & 2 \end{pmatrix}$ - это две матрицы.

Матрицы можно умножать. НО НЕ ТАК, как указано ниже:

$$\begin{pmatrix} 1 & 3 \\ 0 & 5 \end{pmatrix} * \begin{pmatrix} 2 & 3 \\ 1 & 2 \end{pmatrix} = \begin{pmatrix} 1*2 & 3*3 \\ 0*1 & 5*2 \end{pmatrix} = \begin{pmatrix} 2 & 9 \\ 0 & 10 \end{pmatrix}$$

А для этого нужно знать, что такое матрицы

На самом деле матрицы умножаются по правилу

$$\begin{pmatrix}1 & 3\\0 & 5\end{pmatrix} * \begin{pmatrix}2 & 3\\1 & 2\end{pmatrix} = \begin{pmatrix}1*2+3*1 & 1*3+3*2\\0*2+5*1 & 0*3+5*2\end{pmatrix}$$
$$= \begin{pmatrix}5 & 9\\5 & 10\end{pmatrix}$$

Правило умножения позволяет перемножать и неквадратные матрицы. Главное, чтобы строка первой матрицы полностью накладывалась на столбец второй матрицы.

Умножение неквадратных матриц

$$\begin{pmatrix} 5 & 1 \\ 1 & 4 \end{pmatrix} * \begin{pmatrix} 2 & 3 & 0 \\ 0 & 1 & 4 \end{pmatrix} = \begin{pmatrix} 10 & 16 & 4 \\ 2 & 7 & 16 \end{pmatrix}$$

А что если представить нашу таблицу с данными в виде произведения других двух матриц?

	Признаки
Объекты	…

=

	Новые признаки
Объекты	…

*

	Признаки
Новые признаки	…

Причем число новых признаков будет меньше чем старых.

Умножение неквадратных матриц

Первая матрица содержит описание объектов с помощью новых признаков, а вторая матрица содержит описание новых признаков через старые.

Nonnegative matrix factorization (NMF)

Итак, для матрицы A нужно найти матрицы B,C такие, что A=B*C, причем

1) Число столбцов в B должно быть меньше чем в A;
2) Все элементы матриц B,C должны быть неотрицательны.
3) Если таких матриц B,C не существует, то найти матрицы, удовл. пп 1-2, для которых равенство A=B*C выполняется приблизительно.

Это и называется неотрицательным разложением матрицы (NMF). Например,

$$\begin{pmatrix} 10 & 16 & 4 \\ 2 & 7 & 16 \end{pmatrix} = \begin{pmatrix} 5 & 1 \\ 1 & 4 \end{pmatrix} * \begin{pmatrix} 2 & 3 & 0 \\ 0 & 1 & 4 \end{pmatrix}$$

Смысл разложения

$$\begin{pmatrix} 10 & 16 & 4 \\ 2 & 7 & 16 \end{pmatrix} = \begin{pmatrix} 5 & 1 \\ 1 & 4 \end{pmatrix} * \begin{pmatrix} 2 & 3 & 0 \\ 0 & 1 & 4 \end{pmatrix}$$

Это означает, что таблицу с 2-мя объектами и 3-мя признаками можно
представить таблицей с 2-мя признаками (1й множитель), а новые признаки описываются через старые (2й множитель):

	пр1	пр2	пр3
Объект1	10	16	4
Объект2	2	7	16

=

	нов1	нов2
Объект1	5	1
Объект2	1	4

*

	пр1	пр2	пр3
нов1	2	3	0
нов2	0	1	4

При чём тут кластеризация?

Новые признаки можно рассматривать как метки кластеров. То есть вероятность того, что первый объект принадлежит первому кластеру в пять раз выше чем ко второму. А вероятность принадлежности второго объекта второму кластеру в четыре раза выше чем к первому.

	нов1	нов2
Объект1	5	1
Объект2	1	4

Вернемся к задаче о покупателях

К матрице с данными можно применить NMF. Получим

	Мука	Возд.шары	Пиво	Сахар	Чипсы
Покупатель1	0	3	8	0	1
Покупатель2	0	2	5	1	0
Покупатель3	5	0	1	10	0
Покупатель4	0	20	40	2	1
Покупатель5	10	0	1	10	1

=

	нов1	нов2
Пок1	0	1.2850
Пок2	0.4711	0.8065
Пок3	8.4380	0.0365
Пок4	0.0217	6.7563
Пок5	10.847	0

*

	Мука	Возд.шары	Пиво	Сахар	Чипсы
нов1	0.8	0	0.09	1.02	0.06
нов2	0	2.93	5.93	0.29	0.17

Отметим, что это равенство лишь приблизительное. В методе NMF это допускается.

Получаем кластеризацию покупателей

	нов1	нов2
Пок1	0	1.2850
Пок2	0.4711	0.8065
Пок3	8.4380	0.0365
Пок4	0.0217	6.7563
Пок5	10.847	0

Новые признаки из первой таблицы задают кластеризацию покупателей(покупатель относится к i-му кластеру, если число в i-м столбце максимально).
Получаем кластеры {1,2,4},{3,5}.

Смысл новых признаков

Новые признаки здесь имеют очевидную интерпретацию (см. вторую таблицу).
Признак «нов1»=«товары для выпечки».
Признак «нов2»=«товары для праздника».

	Мука	Возд.шары	Пиво	Сахар	Чипсы
нов1	0.8	0	0.09	1.02	0.06
нов2	0	2.93	5.93	0.29	0.17

Кстати, таблица не дает ответа, к какой группе товаров относятся чипсы)))))

Эта техника используется в рекомендательных системах (см. ниже)

Применение NMF
в рекомендательных системах

Рекомендация товаров с помощью NMF

Ранее было получено разложение.

	Мука	Возд.шары	Пиво	Сахар	Чипсы
Покупатель1	0	3	8	0	1
Покупатель2	0	2	5	1	0
Покупатель3	5	0	1	10	0
Покупатель4	0	20	40	2	1
Покупатель5	10	0	1	10	1

=

	нов1	нов2
Пок1	0	1.2850
Пок2	0.4711	0.8065
Пок3	8.4380	0.0365
Пок4	0.0217	6.7563
Пок5	10.847	0

*

	Мука	Возд.шары	Пиво	Сахар	Чипсы
нов1	0.8	0	0.09	1.02	0.06
нов2	0	2.93	5.93	0.29	0.17

Можем ли мы оценить, сколько в будущем потребуется муки и чипсов 2-му покупателю (пока он их не покупал)?

Рекомендация товаров с помощью NMF

Нужно вспомнить, что это не точное равенство.

	Мука	Возд.шары	Пиво	Сахар	Чипсы
Покупатель1	0	3	8	0	1
Покупатель2	0	2	5	1	0
Покупатель3	5	0	1	10	0
Покупатель4	0	20	40	2	1
Покупатель5	10	0	1	10	1

=

	нов1	нов2
Пок1	0	1.2850
Пок2	0.4711	0.8065
Пок3	8.4380	0.0365
Пок4	0.0217	6.7563
Пок5	10.847	0

*

	Мука	Возд.шары	Пиво	Сахар	Чипсы
нов1	0.8	0	0.09	1.02	0.06
нов2	0	2.93	5.93	0.29	0.17

Чтобы получить оценки для товаров, которые человек еще не покупал, то нужно перемножить 2 матрицы справа…

Рекомендация товаров с помощью NMF

Перемножаем и получаем не совсем исходную матрицу

	нов1	нов2
Пок1	0	1.2850
Пок2	0.4711	0.8065
Пок3	8.4380	0.0365
Пок4	0.0217	6.7563
Пок5	10.847	0

	Мука	Возд.шары	Пиво	Сахар	Чипсы
нов1	0.8	0	0.09	1.02	0.06
нов2	0	2.93	5.93	0.29	0.17

=

	Мука	Возд.шары	Пиво	Сахар	Чипсы
Покупатель1	0	3.76505	7.62005	0.37265	0.21845
Покупатель2	0.37688	2.363045	4.824944	0.714407	0.165371
Покупатель3	6.7504	0.106945	0.975865	8.617345	0.512485
Покупатель4	0.01736	19.795959	40.066812	1.981461	1.149873
Покупатель5	8.6776	0	0.97623	11.06394	0.65082

Рекомендация товаров с помощью NMF

Можно сравнить обе матрицы. И хотя 2-й покупатель ни разу не покупал муку и чипсы, мы оцениваем, что муку он «любит» в 2 раза больше чем чипсы

	Мука	Возд.шары	Пиво	Сахар	Чипсы
Покупатель1	0	3	8	0	1
Покупатель2	0	2	5	1	0
Покупатель3	5	0	1	10	0
Покупатель4	0	20	40	2	1
Покупатель5	10	0	1	10	1

	Мука	Возд.шары	Пиво	Сахар	Чипсы
Покупатель1	0	3.76505	7.62005	0.37265	0.21845
Покупатель2	0.37688	2.363045	4.824944	0.714407	0.165371
Покупатель3	6.7504	0.106945	0.975865	8.617345	0.512485
Покупатель4	0.01736	19.795959	40.066812	1.981461	1.149873
Покупатель5	8.6776	0	0.97623	11.06394	0.65082

Использованная литература

1. Т.Сегаран «Программируем коллективный разум» (там пример про кластеризацию новостей)
2. Лекции Воронцова.
3. Википедия «k-means».
4. https://habrahabr.ru/company/ods/blog/325654/