Кластеризация данных

Александр Котов, Николай Красильников

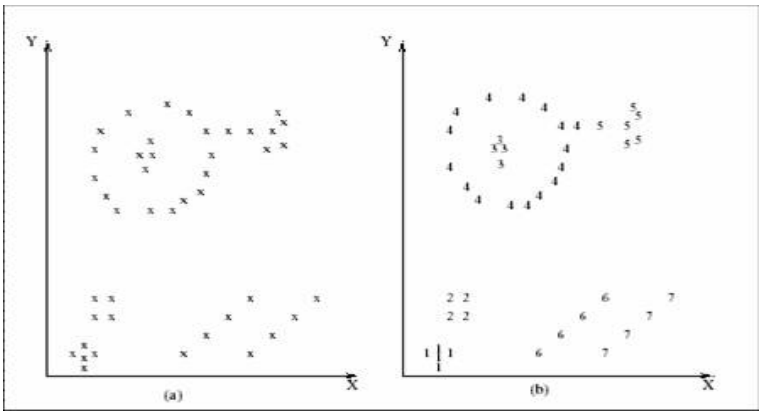
2 октября 2006 г.

Содержание

1 Что такое кластеризация?

Кластеризация - это автоматическое разбиение элементов некоторого множества на группы в зависимости от их схожести. Элементами множества может быть что угодно, например, данные или вектора характеристик. Сами же группы принято также называть кластерами.

1.1 Кластеризация (пример)



2 Зачем это нужно?

У кластеризации существует большое колличество практических применений как в информатике так и в других областях. Примерами применения могут служить:

1. Анализ данных
2. Извлечение и поиск информации

3. Группировка и распознавание объектов

Так же кластеризация сама по себе является важной формой абстракции данных.

Кроме того, кластеризация является бурно развивающимся разделом современной теоретической информатики и в этой области можно получить ряд интересных исследовательских результатов.

3 Формальные определения

Введем определения тех понятий, с которыми будем оперировать.

Объект - элементарная группа данных, с которой оперирует алгоритмы кластеризации.

Каждому объекту отождествляется *вектор характеристик*.

$\mathbf{x} = (x_1, ..., x_d)$

Компоненты x_i являются отдельными *характеристиками* объекта.

Колличество характеристик d определяет *размерность* пространства характеристик.

Множество, состоящее из всех векторов характеристик будем обозначать æ. æ $= (\mathbf{x}_1, ..., \mathbf{x}_n)$, где $\mathbf{x}_i = (x_{i,1}, ..., x_{i,d})$

Кластер - подмножество "близких друг к другу" объектов из æ.

Расстояние $d(\mathbf{x}_i, \mathbf{x}_j)$ между объектами $\mathbf{x}_i$ и $\mathbf{x}_j$ - результат применения выбранной метрики (или квази-метрики) в пространстве характеристик.

3.1 Разница между кластеризацией и классификацией

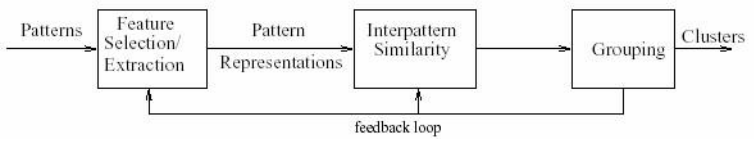
Кластеризация разбивает множество объектов на группы, которые определяются только ее результатом.

Классификация относит каждый объект к одной из заранее определенных групп.

4 Общая схема кластеризации

Кластеризация данных включает в себя следующие этапы:

1. Выделение характеристик
2. Определение метрики
3. Разбиение объектов на группы
4. Представление результатов



Далле каждый из них будет рассмотрен подробно.

5 Выделение вектора характеристик

Для начала необходимо выбрать свойства, которые характеризуют наши объекты. Ими могут быть количественные характеристики (координаты, интервалы...), качественные характеристики (цвет, статус, воинское звание...) и т.д.

Затем стоит попробовать уменьшить размерность пространства характеристических векторов, то есть выделить наиболее важные свойства объектов. Уменьшение размерности ускоряет процесс кластеризации и в ряде случаев позволяет визуфльно оценивать ее результаты.

Выделенные характеристики стоит нормализовать.

Далее все объекты представляются в виде характеристических векторов.

Мы будем полностью отождествлять объект с его характеристическим вектором.

6 Выбор метрики

Следующим этапом кластеризации является выбор метрики, по которой мы будем определять длизость объектов.

Метрика выбирается в зависимости от:

1. пространства, в котором расположены объекты
2. неявных характеристик кластеров

Например, если все координаты объекта непрерывны и вещественны, а кластеры должны представлять собой нечто вроде гиперсфер, то используется классическая метрика Евклида (на самом деле, чаще всего так и есть):

$d_2(x_i, x_j) = (\sum_{k=1}^{d}(x_{i,k} - x_{j,k})^2)^{1/2} = ||x_i - x_j||_2$

7 Алгоритмы кластеризации

Далее мы рассмотрим следующие алгоритмы кластеризации:

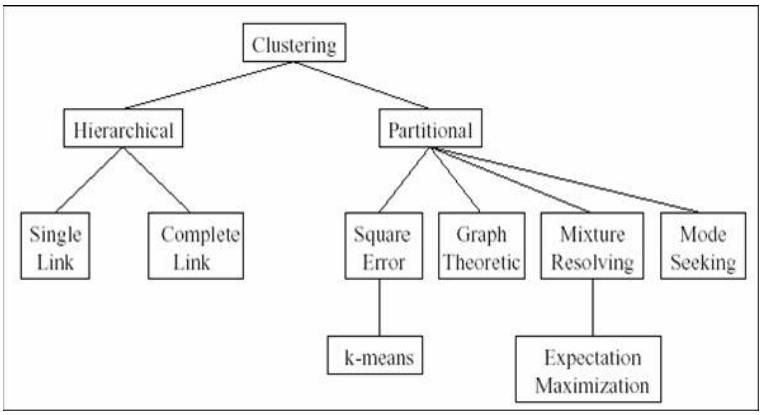
1. Иерархические алгоритмы
2. k-Means алгоритм
3. Минимальное покрывающее дерево
4. Метод ближайшего соседа
5. Алгоритмы нечеткой кластеризации
6. Применение нейронных сетей

7. Генетические алгоритмы
8. Метод закалки

7.1 Классификация алгоритмов

Алгоритмы кластеризации делят на:

- Строящие “снизу-вверх” и “сверху-вниз”
- Монотетические и политетические
- Непересекающиеся и нечеткие
- Детерминированные и стохастические
- Потоковые (online) и не потоковые
- Зависящие и не зависящие от начального разбиения
- Зависящие и не зависящие от порядка рассмотрения объектов



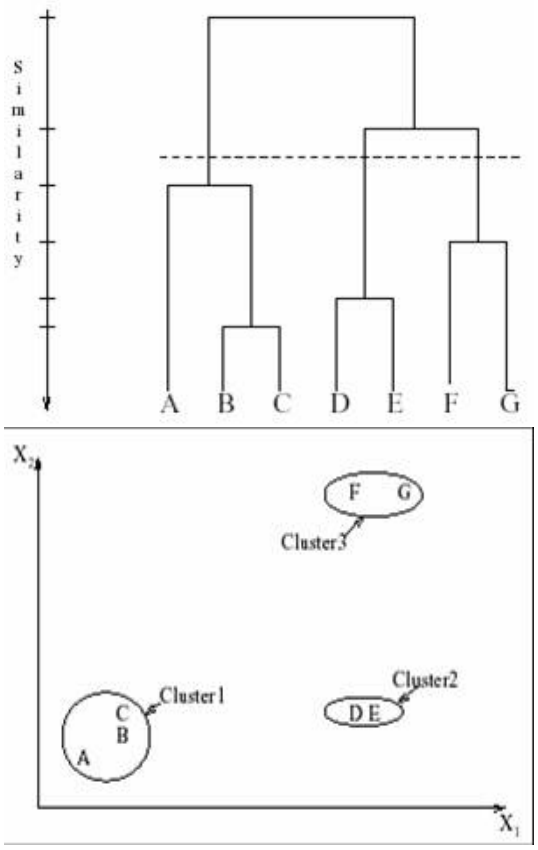
7.2 Иерархические алгоритмы

Результатом работы иерархических алгоритмов является дендограмма (иерархия), позволяющая разбить исходное множество объектов на любое число кластеров.

Два наиболее популярных алгоритма, оба строят разбиение “снизу вверх”:

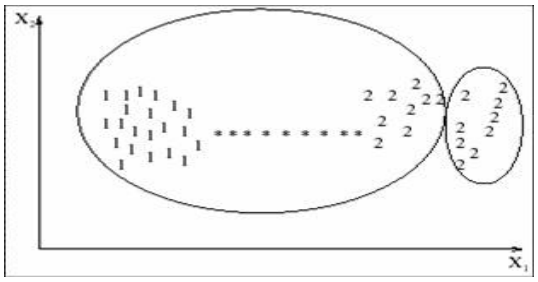
1. *Single-link* - на каждом шаге объединяет два кластера с наименьшим расстоянием между двумя любыми представителями
2. *Complete-link* - на каждом шаге объединяет два кластера с наименьшим расстоянием между двумя наиболее удаленными представителями

7.2.1 Single-link (пример)

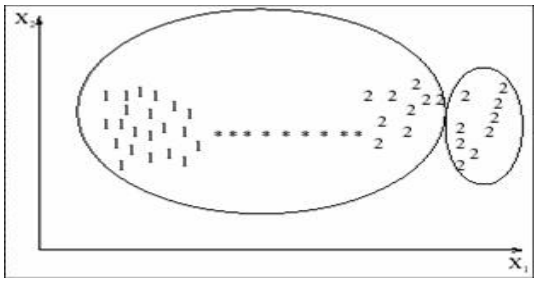


7.2.2 Single-link versus Complete-link

Single-link



Complete-link



7.3 Кластеризация как задача оптимизации

Кластеризацию можно рассмотреть как задачу построения оптимального разбиение объектов на группы.

При этом оптимальность может быть определена как требование минимизации среднеквадратической ошибки разбиения:

$e^2(X, L) = \sum_{j=1}^{K} \sum_{i=1}^{n_j} ||x_i^{(j)} - c_j||^2$, где c_j - "центр масс" кластера j. "Центр масс" кластера - точка в пространстве характеристических векторов со средними для данного кластера значениями характеристик.

7.4 k-Means алгоритм

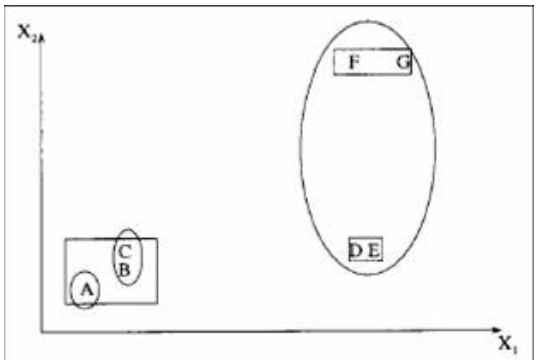
Данный алгоритм состоит из следующих шагов:

1. Случайно выбрать k точек, являющихся начальными "центрами масс" кластеров (любые k из n объектов, или вообще k случайных точек)
2. Отнести каждый объект к кластеру с ближайшим "центром масс"
3. Пересчитать "центры масс" кластеров согласно текущему членству
4. Если критерий остановки алгоритма не удовлетворен, вернуться к шагу 2

В качестве критерия остановки обычно выбирают один из двух:

1. Отсутствие перехода объектов из кластера в кластер на шаге 2
2. Минимальное изменение среднеквадратической ошибки

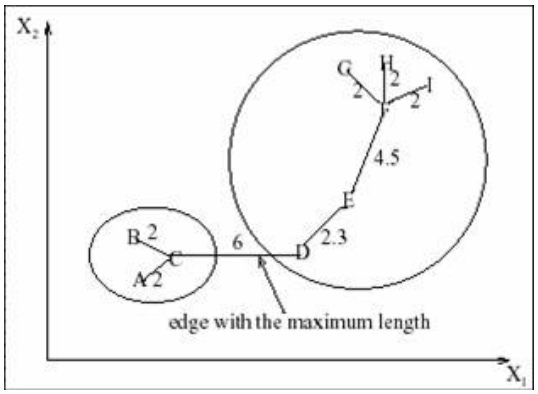
Алгоритм чувствителен к начальному выбору "центров масс":



7.5 Минимальное покрывающее дерево

Данный метод производит иерархическую кластеризацию “сверху-вниз”.

Сначала все объекты помещаются в один кластер. Затем на каждом шаге один из кластеров разбивается на два, так чтобы расстояние между ними было максимальным.



7.6 Метод ближайшего соседа

Этот метод является одиним из старейших методов кластеризации. Он был создан 1978 году. Он прост и наименее оптимален из всех представленных в данной лекции.

Для каждого объекта вне кластера делаем следующее:

1. Находим его ближайшего соседа, кластер которого определен.
2. Если расстояние до этого соседа меньше порога, то относим его в тот же кластер. Иначе из рассматриваемого объекта создается еще один кластер.

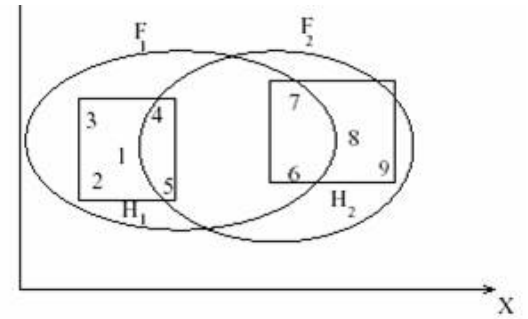
Далее рассматривается результат и при необходимости увеличивается порог. Например, если много кластеров из одного объекта.

7.7 Нечеткая кластеризация

Четкая (непересекающаяся) кластеризация - кластеризация, которая каждый $\mathbf{x}_i$ из æ относит только одному кластеру.

Нечеткая кластеризация - кластеризация, при которой для каждого $\mathbf{x}_i$ из æ определяется $f_{i,k}$. $f_{i,k}$ - вещественное значение, показывающее степень принадлежности $\mathbf{x}_i$ к кластеру j.

Пример:



$F_1 = \{(1, 0.9), (2, 0.8), (3, 0.7), (4, 0.6), (5, 0.55), (6, 0.2), (7, 0.2), (8, 0.0), (9, 0.0)\}$
$F_2 = \{(1, 0.0), (2, 0.0), (3, 0.0), (4, 0.1), (5, 0.15), (6, 0.4), (7, 0.35), (8, 1.0), (9, 0.9)\}$

7.7.1 Алгоритм нечеткой кластеризации

Алгоритм следующий:

1. Выбрать начальное нечеткое разбиение n объектов на k кластеров путем выбора матрицы принадлежности U размера nxk. Обычно $U_{ij} \in [0, 1]$.

2. Используя матрицу U, найти значение критерия нечеткой ошибки. Например, $E^2(X, U) = \sum_{i=1}^{N} \sum_{k=1}^{K} U_{ik} ||x_i^{(j)} - c_k||^2$, где c_k - "центр масс" нечеткого кластера k, $c_k = \sum_{i=1}^{N} U_{ik} x_i$.

3. Перегруппировать объекты с целью уменьшения этого значения критерия нечеткой ошибки.

4. Возвращаться в пункт 2 до тех пор, пока изменения матрицы U не станут незначительными.

7.8 Применение нейронных сетей

Порой для решения задач кластеризации применяются нейронные сети. У данного подхода есть следующие особенности:

- Искусственные нейронные сети легко работают в распределенных системах с большой параллелизацией в силу своей природы.

- Искусственные нейронные сети оперируют числами, поэтому они могут проводить разбиение на кластеры только для объектов с численными векторами характеристик.
- Поскольку искусственные нейронные сети подстраивают свои весовые коэффициенты, основываясь на исходных данных, это помогает сделать выбор значимых характеристик (этап 1 кластеризации) менее субъективным.

Примером может служить кластеризация с применением самоорганизующихся карт Кохонена. Она является аналогом алгоритма k-Means.

7.9 Генетические алгоритмы

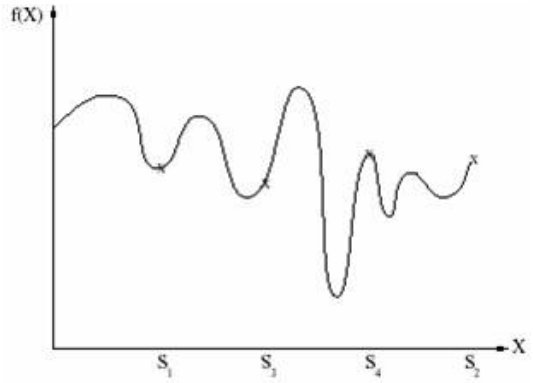
Общая схема данного подхода:

1. Выбрать начальную случайную популяцию множества решений и получить оценку качества для каждого решения (обычно она пропорциональна $1/e^2$)
2. Создать и оценить следующую популяцию решений, используя эволюционные операторы: Оператор выбора - с большей вероятностью предпочитает хорошие решения Оператор рекомбинации (обычно это “кроссовер”) - создает новое решение на основе рекомбинации из существующих Оператор мутации - создает новое решение на основе случайного незначительного изменения одного из существующих
3. Повторять шаг 2 до получения нужного результата

Главным достоинством генетических алгоритмов в данном применении является то, что они ищут глобальное оптимальное решение.

Большинство популярных алгоритмов оптимизации выбирают начальное решение, которое затем изменяется в ту или иную сторону. Таким образом получается хорошее разбиение, но не всегда - самое оптимальное.

Операторы рекомбинации и мутации позволяют получить решения, сильно не похожие на исходные - таким образом осуществляется глобальный поиск



7.10 Метод закалки

Этот метод также пытается найти глобальный оптимум, однако этот метод работает только с одним текущим решением.

1. Случайно выбрать начальное разбиение P_0, сосчитать для него ошибку E_{P_0}. Выбрать значения для контрольных параметров - начальной T_0 и конечной T_f температур ($T_0 > T_f$)
2. Выбрать разбиение P_1 неподалеку от P_0 и сосчитать E_{P_1}. Если $E_{P_0} > E_{P_1}$, то сделать текущим разбиение P_1, иначе - сделать текущим разбиение P_1 с вероятностью, зависящей от разницы температур. Повторить выбор соседних разбиений несколько раз.
3. Чуть-чуть "остыть": $T_0 = c * T_0$, (c - заранее определенная константа, $c < 1$). Если $T_0 > T_f$ перейти к шагу 2, иначе закончить работу.

7.11 Какой алгоритм выбрать?

При выборе алгоритма полезно учитывать следующее:

- Генетические алгоритмы и искусственные нейронные сети хорошо распараллеливаются
- Генетические алгоритмы и метод закалки осуществляют глобальный поиск
- Генетические алгоритмы хорошо работают только для одно- (двух-) мерных объектов, зато не требуется непрерывность координат
- k-Means быстро работает и прост в реализации, но дает только гиперсферические кластеры
- Иерархические алгоритмы дают оптимальное разбиение на кластеры, но их трудоемкость квадратична
- На практике лучше всего зарекомендовали себя гибридные подходы, где шлифовка кластеров выполняется методом k-Means, а первоначальное разбиение - одним из более сильных методов

7.12 Априорное использование природы кластеров в алгоритмах

- Неявное использование:
  - выбор соответствующих характеристик объектов из всех характеристик
  - выбор метрики (метрика Евклида обычно дает гиперсферические кластеры)
- Явное использование:
  - подсчет схожести (использование ∞ для расстояния между объектами из заведомо разных кластеров)
  - представление результатов (учет явных ограничений)

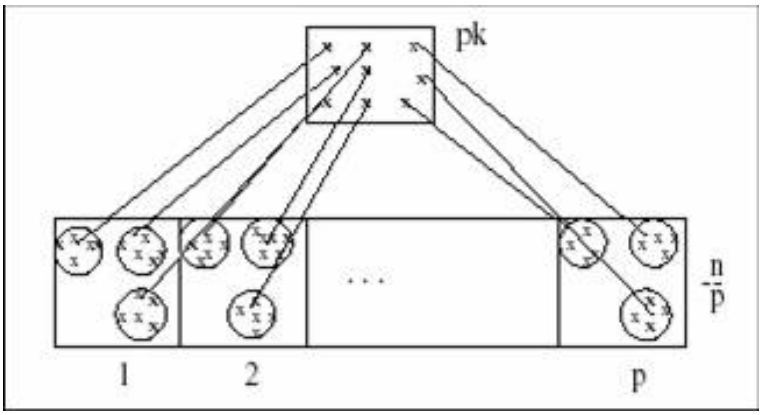
7.13 Кластеризация больших объемов данных

При кластеризации больших объемов данных обычно используют k-Means или его гибридные модификации.

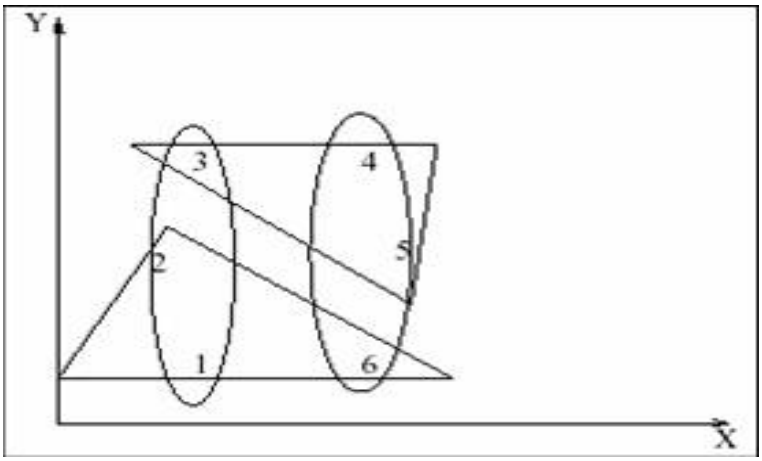
Если множество объектов не помещается в основную память, можно:

- проводить кластеризацию по принципу “разделяй и властвуй”, путем извлечения подмножеств, проведения кластеризации внутри них и последующей работой с только одним представителем каждого кластера
- использовать потоковые (on-line) алгоритмы (например, leader, модификация метода ближайшего соседа)
- использовать параллельные вычисления

7.13.1 Разделяй и властвуй (пример)



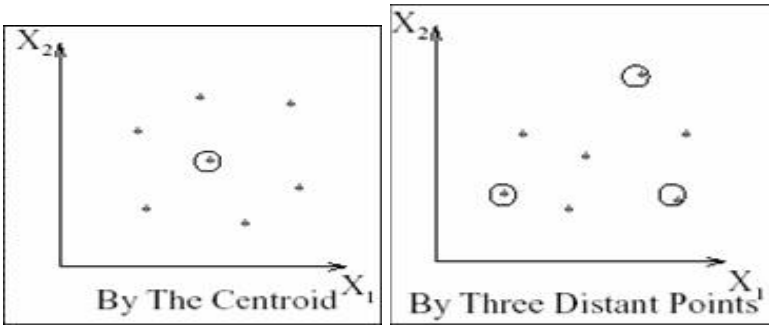
7.13.2 Алгоритм Leader (пример)



8 Представление результатов

Результаты кластеризации должны быть представлены в удобном для обработки виде. Обычно используется один из следующих способов:

- представление кластеров центроидами
- представление кластеров набором характерных точек
- представление кластеров их ограничениями



9 Применения кластеризации

- Анализ данных (Data mining)
  - Упрощение работы с информацией
  - Визуализация данных
- Группировка и распознавание объектов
  - Распознавание образов
  - Группировка объектов
- Извлечение и поиск информации
  - Построение удобных классификаторов

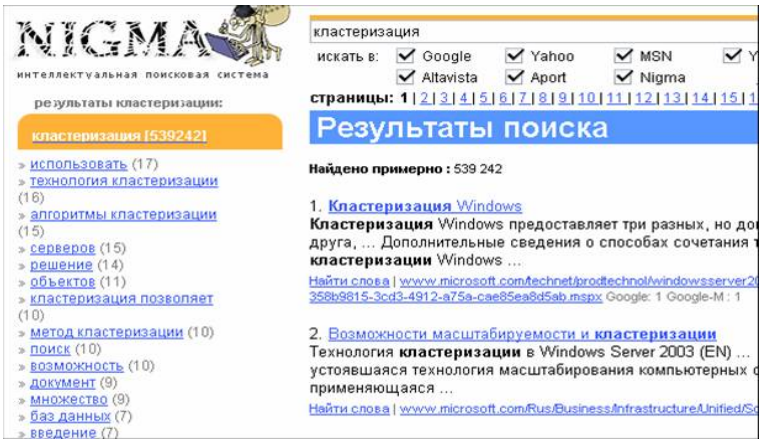
9.1 Анализ данных (Data mining)

Использование кластеризации упрощает работу с информацией, так как:

- достаточно работать только с k представителями кластеров;
- легко найти "похожие" объекты - такой поиск применяется в ряде поисковых движков (http://www.nigma.ru, http://www.vivisimo.com, ...);
- происходит автоматическое построение каталогов.

Также наглядное представление кластеров позволяет понять структуру множества объектов æ в пространстве.

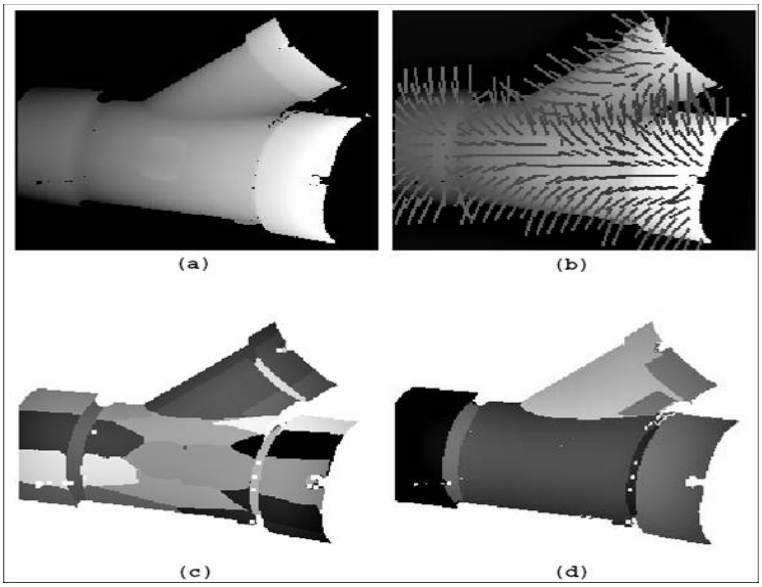
9.1.1 http://www.nigma.ru (пример)



9.2 Группировка и распознавание объектов

- Распознавание образов (OCR и др.)
  - Построение кластеров на основе большого набора учебных данных
  - Присвоение каждому из кластеров соответствующей метки
  - Ассоциирование каждого объекта, полученного на вход алгоритма распознавания, с меткой соответствующего кластера
- Группировка объектов
  - Сегментация изображений
  - Уменьшение количества информации

9.2.1 Сегментация изображений (пример)



9.3 Извлечение и поиск информации (на примере книг в библиотеке)

Наиболее известная система не автоматической классификации - LCC (Library of Congress Classification)

- Метка Q означает книги по науке
- Подкласс QA - книги по математике
- Метки с QA76 до QA76.8 - книги по теоретической информатике

Проблемы LCC:

- LCC относит каждую книгу только к одной категории
- Иногда классификация отстает от быстрого развития некоторых областей науки

Автоматическая кластеризация приходит на помощь:

- Нечеткое разбиение на группы решает проблему одной категории
- Новые кластеры вырастают одновременно с развитием области

10 Итого

- Кластеризация - это автоматическое разбиение множества объектов на группы по принципу схожести
- Общая схема кластеризации одна (выделение характеристик $\mapsto$ выбор метрики $\mapsto$ группировка объектов $\mapsto$ представление результатов). Но существует много реализаций этой схемы.
- Кластеризация данных широко применяется в современной информатике.

11 Источники

1. A.K. Jain, M.N. Murty, P.J. Flynn - "Data Clustering: A Review" (http://www.csee.umbc.edu/ nicholas/clustering/p264-jain.pdf)
2. J. Kogan, C. Nicholas, M. Teboulle - "Clustering Large and High Dimensional data" (http://www.csee.umbc.edu/ nicholas/clustering/tutorial.pdf)