МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ТУЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт *Прикладной математики и компьютерных наук*

Кафедра *Информационной безопасности*

СБОРНИК МЕТОДИЧЕСКИХ УКАЗАНИЙ К ЛАБОРАТОРНЫМ РАБОТАМ

по дисциплине

МЕТОДЫ АНАЛИЗА ДАННЫХ

Направление подготовки: 09.04.01 *«Информатика и вычислительная техника»*
Профиль : *«Компьютерный анализ и интерпретация данных»*

Квалификация (степень) выпускника: *магистр*

Формы обучения: *очная*

Тула 2015 г.

Методические указания к лабораторным работам учебной дисциплины *(модуля) «Методы анализа данных»* разработаны *проф. С.Д. Двоенко* и обсуждена на заседании кафедры Информационной безопасности института Прикладной математики и компьютерных наук (протокол заседания кафедры №____ от «____» _____ 20___ г.)

Разработчик(и) МУ ЛР дисциплины *(модуля)* _____.

личная подпись(и)

Лабораторная работа №1

ПРЕДСТАВЛЕНИЕ ДАННЫХ

Цель и задача работы

Приведение экспериментальных данных к стандартизованному виду. Преобразования матрицы данных.

Теоретические положения

МАТРИЦА ДАННЫХ

Рассмотрим традиционный вид представления результатов эксперимента - матрицу данных. Пусть исследователь располагает совокупностью из N наблюдений над состоянием исследуемого явления. Пусть при этом явление описано набором из n характеристик, значения которых тем или иным способом измерены в ходе эксперимента. Данные характеристики носят название признаков, показателей или параметров. Такая информация представляется в виде двухмерной таблицы чисел $\mathbf{X}$ размерности $N \times n$ или в виде матрицы $\mathbf{X}(N \times n)$:

$$\begin{array}{c} \\ \mathbf{x}_1 \\ \cdot \\ \mathbf{x}_i \\ \cdot \\ \mathbf{x}_N \end{array} \begin{array}{c} \begin{array}{ccccc} X_1 & \dots & X_j & \dots & X_n \end{array} \\ \begin{pmatrix} x_{11} & \dots & x_{1j} & \dots & x_{1n} \\ \cdot & \dots & \cdot & \dots & \cdot \\ x_{i1} & \dots & x_{ij} & \dots & x_{in} \\ \cdot & \dots & \cdot & \dots & \cdot \\ x_{N1} & \dots & x_{Nj} & \dots & x_{Nn} \end{pmatrix} \end{array}.$$

Строки матрицы $\mathbf{X}$ соответствуют наблюдениям или, другими словами, объектам наблюдения. В качестве объектов наблюдения выступают, например, в социологии - респонденты (анкетируемые люди), в экономике - предприятия, виды продукции и т.д. Столбцы матрицы $\mathbf{X}$ соответствуют признакам, характеризующим изучаемое явление. Как правило, это наиболее легко измеряемые характеристики объектов. Например, предприятие характеризуется численностью, стоимостью основных фондов, видом выпускаемой продукции и т.д. Очевидно, что элемент x_{ij} представляет собой значение признака j, измеренное на объекте i.

Часто матрица данных приводится к стандартной форме преобразованием

$$x'_{ij} = \left(x_{ij} - \bar{x}_j\right) \Big/ \sigma_j, \quad \bar{x}_j = \frac{1}{N}\sum_{i=1}^{N} x_{ij}, \quad \sigma_j^2 = \frac{1}{N}\sum_{i=1}^{N}\left(x_{ij} - \bar{x}_j\right)^2, \quad i = 1,\dots N;\ j = 1,\dots n,$$

где $\bar{x}_j, \sigma_j^2 -$ среднее и дисперсия по столбцу j, после которого стандартная матрица $\mathbf{X}'$ обладает свойствами

$$\bar{x}'_j = \frac{1}{N}\sum_{i=1}^{N} x'_{ij} = 0, \qquad \sigma'^2_j = \frac{1}{N}\sum_{i=1}^{N} x'^{\,2}_{ij} = 1, \quad i = 1,\dots N;\ j = 1,\dots n.$$

В дальнейшем будем использовать для матрицы данных обозначение $\mathbf{X}$, полагая, что это стандартизованная матрица, без дополнительного упоминания. Для пояснения заметим, что часто признаки, описывающие некоторый объект, имеют существенно различный физический смысл. Это приводит к тому, что величины в различных столбцах исходной матрицы трудно сопоставлять между собой, например, *кг* и *м*. Поэтому

получение стандартизованной матрицы можно понимать как приведение всех признаков к некоторой единой условной физической величине, измеренной в одних и тех же условных единицах.

ГИПОТЕЗА КОМПАКТНОСТИ

Рассмотрим *n*-мерное пространство, где оси координат соответствуют отдельным признакам матрицы данных **X**. Тогда каждую строку матрицы данных можно представить как вектор в этом пространстве. Следовательно, каждый из *N* объектов наблюдения представлен своей изображающей точкой в *n*-мерном пространстве признаков.

Отметим, что в основе различных методов анализа матрицы данных лежит неформальное предположение, условно названное “гипотезой компактности”. Предполагается, что объекты наблюдения в различной степени “похожи” друг на друга. Предполагается, что все множество большого числа объектов представимо в виде небольшого числа достаточно сильно различающихся подмножеств, внутри которых объекты наблюдения “сильно похожи”. Например, сильно различающиеся подмножества характеризуют типы различных состояний изучаемого явления, а похожие объекты внутри них являются зафиксированными состояниями явления, где разброс значений объясняется ошибками измерения, изменением условий эксперимента и т.д.

Такие компактные множества называются классами, кластерами, таксонами. При справедливости такой гипотезы задача обработки в наиболее общей формулировке неформально ставится как задача разбиения исходного множества объектов в признаковом пространстве на конечное число классов. Не вдаваясь глубоко в суть различных постановок задачи классификации, отметим следующие важные моменты.

Во-первых, при известном числе классов, как правило, требуется получить наиболее удаленные друг от друга в пространстве признаков компактные классы.

Во-вторых, часто число классов заранее неизвестно, поэтому нужно его определить, исходя из априорных соображений, или, пробуя разные варианты разбиения на классы.

В-третьих, важно, чтобы результат разбиения был устойчивым. Например, методы, используемые в одном из направлений обработки данных - кластер-анализе - могут порождать различные разбиения для небольших изменений матрицы данных. Так, если в исходную матрицу добавить новые объекты, то результат кластеризации изменится. Если он изменится незначительно по составу кластеров, удаленности кластеров друг от друга, их размеру в пространстве, то результат можно считать устойчивым.

В-четвертых, другие методы классификации, например, в распознавании образов, направлены не на получение таксономии (перечисление принадлежности объектов каждому из классов), а на получение способа определять класс каждого добавляемого к матрице данных объекта. Данный метод реализуется в виде так называемого решающего правила. Оно представляет собой функцию $g(\mathbf{x})$, принимающую значения на конечном множестве из m классов $\{\Omega_1,\dots\Omega_m\}$. Тогда при предъявлении объекта $\mathbf{x}\in\Omega_i$, решающая функция примет значение $g(\mathbf{x})=\Omega_i$.

Заметим, что разбиение объектов наблюдения на классы означает разделение матрицы данных на горизонтальные полосы, т.е. перегруппировку строк матрицы так, что внутри каждой из групп строк объекты принадлежат одному классу и не принадлежат другим классам.

С другой стороны, можно рассмотреть *N*-мерное пространство, оси которого соответствуют отдельным объектам. Тогда каждый столбец X_j матрицы **X** представляет собой вектор в данном пространстве, а вся матрица - совокупность *n* векторов.

Такое пространство называется пространством объектов. В нем все векторы X_j одинаковы по длине, вычисляемой как евклидова норма

$$\|X_j\| = \sqrt{\sum_{i=1}^{N} x_{ij}^2} = \sqrt{N\sigma_j^2} = \sqrt{N} .$$

Тогда характеристикой близости признаков X_i и X_j в таком пространстве служит близость направлений их векторов, измеряемая $\cos\alpha_{ij}$, где α_{ij} - угол между ними. В этом смысле векторы близки, если угол между ними близок к нулю или к 180^0, и, следовательно, косинус угла близок по модулю к единице. Равенство $\cos\alpha_{ij}$ по модулю единице означает совпадение векторов и линейную связь, так как в стандартизованной матрице данных значения по одному признаку в точности соответствует значениям по другому признаку, или совпадение векторов с точностью до наоборот, то есть противоположные направления, и, следовательно, также линейную связь. Тогда перпендикулярные векторы и нулевое значение косинуса угла между ними соответствуют наиболее далеким признакам. В этом случае можно предположить противоположную ситуацию, когда признаки наименее зависимы друг от друга - линейно независимы.

Из теории вероятностей и математической статистики известно, что линейная связь между двумя переменными характеризуется коэффициентом корреляции. Случаю двух переменных, где значения каждой из них представлены в виде ряда наблюдений, соответствует выбор двух столбцов и $X_j = \left(x_{1j},\ldots x_{Nj}\right)^T$ в матрице данных. Коэффициент корреляции есть просто скалярное произведение двух векторов признаков в пространстве объектов, нормированное к их длине, то есть просто косинус угла между стандартизованными векторами:

$$r_{ij} = \frac{1}{N}\sum_{k=1}^{N} x_{ki}x_{kj} = \frac{1}{N} X_i^T X_j = \frac{1}{N}\|X_i\|\cdot\|X_j\|\cos\alpha_{ij} = \cos\alpha_{ij} .$$

В статистическом смысле корреляционная связь означает, что значения одного признака имеют тенденцию изменяться синхронно значениям другого признака. Отсутствие связи означает, что изменение значений одного признака никак не сказывается на изменении значений другого признака. Такие признаки считаются статистически независимыми и, в частности, при отсутствии корреляционной связи, линейно независимыми.

Отметим, что в основе понятия о взаимосвязи между признаками лежит неформальное предположение, условно названное “гипотезой скрытых факторов”. А именно, предполагается, что состояние некоторого изучаемого явления определяется “скрытым”, “существенным” фактором, который нельзя измерить непосредственно. Можно лишь измерить набор некоторых других признаков, косвенно отражающих состояние скрытого фактора. Предполагается также, что множество скрытых факторов невелико и значительно меньше набора измеряемых признаков. Тогда группа признаков, испытывающая преимущественное влияние некоторого из факторов, будет более или менее синхронно изменять свои значения при изменении состояния этого скрытого фактора. Чем сильнее влияние скрытого фактора, тем синхроннее меняют свои значения признаки, тем сильнее связь.

В пространстве объектов это означает, что векторы признаков образуют достаточно компактную группу, в которой пучок направлений векторов можно охватить некоторым выпуклым конусом с острой вершиной в начале координат.

При справедливости гипотезы о факторах задача обработки в наиболее общей формулировке неформально ставится как задача выделения конечного числа групп наиболее сильно связанных между собой признаков и построения для каждой из них (либо выбора среди них) одного, наиболее сильно связанного с ними (наиболее близкого к ним) признака, который считается фактором данной группы. Успешное решение такой задачи означает, что в основе сложных взаимосвязей между внешними признаками лежит относительно более простая скрытая структура, отражающая наиболее характерные

и часто повторяющиеся взаимосвязи.

Отметим следующие важные моменты.

Во-первых, различные методы выделения скрытых факторов объединены в группу методов - факторный анализ. Сюда же многие исследователи относят и метод главных компонент.

Во-вторых, существенным в этих методах является то, что число найденных факторов k должно быть много меньше числа признаков n, а найденные факторы должны быть как можно более ортогональны друг другу.

В-третьих, как правило, система факторов должна быть ориентирована так, чтобы факторы были упорядочены по масштабу разброса значений объектов на их осях. В статистических терминах это означает, что факторы должны быть упорядочены по дисперсии объектов на их осях. Необходимость получения именно такой конфигурации объясняется следующим обстоятельством. Возьмем в пространстве факторов главный фактор - фактор с наибольшей дисперсией объектов по его оси. Очевидно, что чем больше дисперсия значений объектов по его оси, тем легче выделить локальные сгущения значений и интерпретировать их как группы похожих объектов, то есть классифицировать их. Такое же предположение применимо и к оставшимся факторам. Если система факторов ортогональна или близка к ней, то факторы считаются независимыми. Тогда разброс значений по оси каждого из факторов можно объяснить влиянием только этого фактора.

Пусть, например, ряды наблюдений двух случайных величин $X_i = \left(x_{1i},\ldots x_{Ni}\right)^T$ и $X_j = \left(x_{1j},\ldots x_{Nj}\right)^T$ являются выборками из генеральной совокупности с нормальным законом распределения. Изобразим пространство двух признаков в виде плоскости с осями координат X_i и X_j (Рис. 1.3).

Плотность вероятности нормального распределения по оси каждого признака есть $f(x) = \left(1/\sigma\sqrt{2\pi}\right)\exp\left[-\left(x-\bar{x}\right)^2/2\sigma^2\right] = \left(1/\sqrt{2\pi}\right)\exp\left(-x^2/2\right)$ при $\bar{x}=0, \sigma=1.$

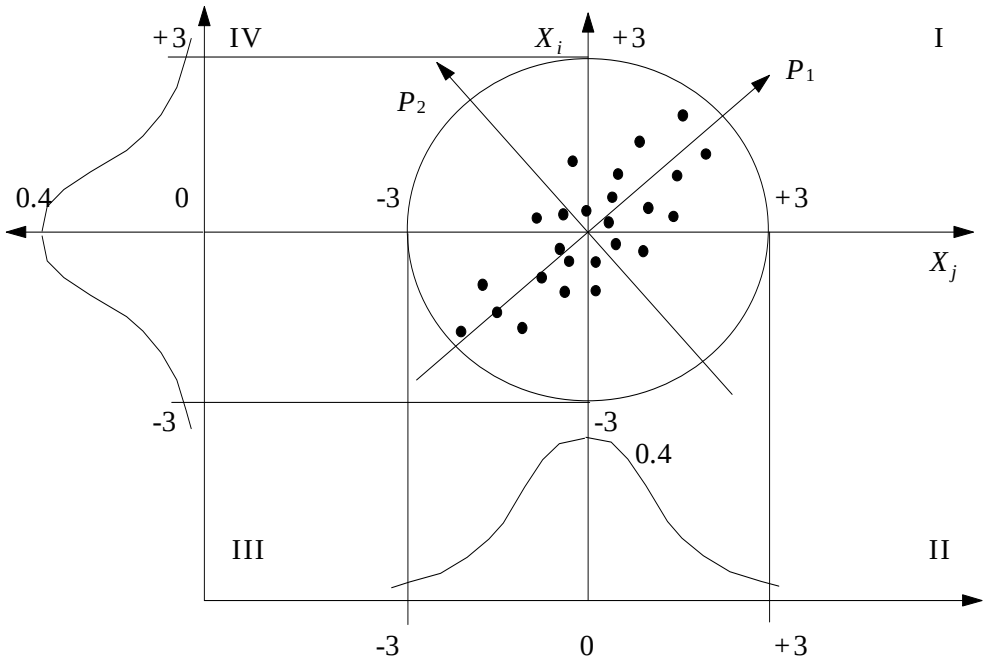


Рис. 1.3. Распределение наблюдений на плоскости.

Согласно хорошо известному правилу “трех сигм”, 99.73% наблюдений нормально распределенной случайной величины попадет в интервал значений по оси аргумента от -3σ до $+3\sigma$, или при $\sigma=1$ от -3 до +3. Следовательно, на плоскости в координатах X_i и

X_j все 99.73% наблюдений будут сосредоточены внутри окружности радиуса 3. При наличии корреляционной связи между признаками наблюдения будут сосредоточены внутри эллипса рассеивания. Чем сильнее окажется связь, тем уже будет эллипс рассеивания. В случае положительной связи, изображенной на рисунке, большие значения одного признака имеют тенденцию соответствовать большим значениям другого признака и наоборот. Поэтому, в большинстве случаев совместные наблюдения значений этих признаков более часто попадают в I и III квадранты плоскости и реже - во II и IV. Кривые равных вероятностей имеют форму вложенных эллипсов с двумя осями P_1 и P_2. Из рисунка легко заметить, что проекции изображающих точек на горизонтальную ось X_j в среднем расположены более плотно, чем проекции тех же точек на ось P_1. Математически доказан факт, что проекции точек на главную ось P_1 эллипса рассеивания расположены наименее плотно по сравнению с другими возможными положениями оси. Если кластеры представляют собой локальные сгущения в эллипсе рассеивания, то переход к системе координат P_1 и P_2 дает наилучшую возможность для их выделения. При достаточно сильной корреляции исходных признаков новый признак P_1 может быть выбран в качестве их фактора.

Заметим, что разбиение признаков на группы означает разбиение матрицы данных на вертикальные полосы, то есть перегруппировку столбцов матрицы так, что внутри одной группы признаки сильно связаны между собой и слабо связаны с любым признаком из другой группы.

МАТРИЦА ОБЪЕКТ-ОБЪЕКТ И ПРИЗНАК-ПРИЗНАК. РАССТОЯНИЕ И БЛИЗОСТЬ

Пусть имеется матрица данных $\mathbf{X}(N \times n)$. Если рассматривать строки данной матрицы как N векторов $\mathbf{x}_i$ в пространстве n признаков, то естественно рассмотреть расстояние между двумя некоторыми векторами. Расстояния между всевозможными парами векторов дают матрицу $\mathbf{R}(N \times N)$ расстояний типа объект - объект.

Напомним, что расстоянием между векторами в пространстве признаков называется некоторая положительная величина d, удовлетворяющая следующим трем аксиомам метрики:

1. $d(\mathbf{x}_1, \mathbf{x}_2) > 0, \quad d(\mathbf{x}_1, \mathbf{x}_1) = 0$;
2. $d(\mathbf{x}_1, \mathbf{x}_2) = d(\mathbf{x}_2, \mathbf{x}_1)$;
3. $d(\mathbf{x}_1, \mathbf{x}_2) + d(\mathbf{x}_2, \mathbf{x}_3) \geq d(\mathbf{x}_1, \mathbf{x}_3)$ (неравенство треугольника).

Таким образом, матрица расстояний является симметричной с нулевой главной диагональю. Существуют различные метрики, но наиболее известной вообще и наиболее применяемой в обработке данных, в частности, является евклидова метрика

$$d(\mathbf{x}_1, \mathbf{x}_2) = \sqrt{\sum_{i=1}^{n} (x_{1i} - x_{2i})^2} .$$

Часто используется линейная метрика вида

$$d(\mathbf{x}_1, \mathbf{x}_2) = \sum_{i=1}^{n} |x_{1i} - x_{2i}| .$$

Применение линейной метрики оправдано, когда расстояние определяется как расстояние между домами в городе по кварталам, а не напрямик. Возможны и другие виды расстояний.

Часто рассматривается величина, обратная в некотором смысле расстоянию - близость. На практике часто используют функции близости вида

$$\mu(\mathbf{x}_1,\mathbf{x}_2)=\exp\left[-\alpha d^2(\mathbf{x}_1,\mathbf{x}_2)\right] \text{ или } \mu(\mathbf{x}_1,\mathbf{x}_2)=\frac{1}{1+\alpha d(\mathbf{x}_1,\mathbf{x}_2)},$$

где α определяет крутизну функции близости. Очевидно, что матрица близостей также является симметричной с единичной главной диагональю, так как $\mu(\mathbf{x}_1,\mathbf{x}_1)=1$.

Если рассмотреть признаки как *n* векторов в *N*-мерном пространстве объектов, то получим другое преобразование матрицы данных в матрицу $\mathbf{R}(n\times n)$ типа признак - признак. Элементом r_{ij} такой матрицы является значение расстояния или близости между признаками X_i и X_j. Наиболее распространено представление в виде матрицы близостей между признаками, где под близостью понимается, например, корреляция соответствующих признаков.

Задание на работу

1. Выбрать матрицу данных в одном из публичных репозиториев данных:

- http://polygon.machinelearning.ru - репозиторий данных и алгоритмов «Полигон» ВЦ РАН

- http://archive.ics.uci.edu/ml/– репозиторий данных Центра машинного обучения и интеллектуальных систем (университет Калифорнии, Ирвайн)

2. Изучить описание данных

3. Составить матрицу количественных данных вида объект-признак
4. Привести матрицу данных к стандартизированному виду

Содержание отчета

Номер и название лабораторной работы;
Цель лабораторной работы;
Доклад к презентации.
Выводы.

Контрольные вопросы

1. Как вычислить коэффициент корреляции
2. Что характеризует коэффициент корреляции
3. Для чего выполняется стандартизация данных
4. В чем заключаются свойства расстояния

Лабораторная работа №2

ИЗУЧЕНИЕ АЛГОРИТМА K-MEANS

Цель и задача работы

Изучение основных методов построения алгоритмов кластер-анализа. Изучение принципа работы алгоритма.

Теоретические положения

АЛГОРИТМЫ КЛАСТЕР- АНАЛИЗА

Решение задачи кластер-анализа также направлено на выявление локальных сгущений объектов в признаковом пространстве. Так как решение задачи кластер-анализа направлено на получение непосредственной классификации перечислением классов объектов, то при ее решении отсутствует этап обучения и не строится решающее правило. Тогда решение задачи кластер-анализа заключается в следующем:

1. выбрать меру близости (расстояния) в пространстве;

2. конструктивно определить понятие локального сгущения;

3. решить проблему классов.

Заметим, что каждая из указанных проблем достаточно нетривиальна сама по себе, и при их решении существует масса тонкостей, как теоретических, так и эмпирических. В то же время интуитивный смысл каждой проблемы достаточно прозрачен. Поэтому далее рассмотрим наиболее очевидные примеры решения данных проблем.

Некоторые наиболее типичные меры близости и расстояния уже были рассмотрены нами ранее. Заметим, что при конструировании таких мер открывается широкое поле деятельности для учета всех особенностей данных, отражающих по мнению исследователя суть задачи обработки.

Конструктивное определение понятия локального сгущения означает, что задается некоторая эвристическая процедура выделения классов, либо задается некоторый критерий и, как правило, итерационная процедура его экстремизации. Например, одним из распространенных показателей качества кластеризации является среднее дисперсий классов, взвешенных по объемам классов, которое нужно минимизировать:

$$J(\mathbf{z}_1,\dots \mathbf{z}_K) = \frac{1}{N}\sum_{i=1}^{K}\sum_{\mathbf{x}\in\Omega_i}\|\mathbf{x}-\mathbf{z}_i\|^2, \quad \mathbf{z}_i = \frac{1}{N_i}\sum_{\mathbf{x}\in\Omega_i}\mathbf{x},$$

где K- заранее заданное число классов Ω_i, $\mathbf{z}_i$- среднее по классу (центр класса), N_i- число объектов в классе Ω_i, N- всего объектов.

Рассмотрим типичный алгоритм минимизации критерия J для заданного числа классов K. Различные модификации данного алгоритма известны как алгоритм K- внутригрупповых средних, алгоритм K - средних, алгоритм Мак-Куина, алгоритм Хартигана.

Шаг 0: Для заданного числа K классов выбираются K исходных центров $\mathbf{z}_i^0, i=1,\dots K$, например, как K самых удаленных друг от друга объектов.

Шаг s: 1. Все множество объектов $\mathbf{x}_l$, $l=1,\dots N$ распределяется по K клас-

сам по правилу $\mathbf{x}\in\Omega_i^s$, если $\left\|\mathbf{x}-\mathbf{z}_i^s\right\|\le\left\|\mathbf{x}-\mathbf{z}_j^s\right\|$, $\quad j=1,\dots K,\quad j\ne i$.

2. Пересчитываются центры классов $\mathbf{z}_i^{s+1}=\dfrac{1}{N_i^s}\sum\limits_{\mathbf{x}\in\Omega_i^S}\mathbf{x}$, $i=1,\dots K$.

3. Если выполнено $\mathbf{z}_i^{s+1}=\mathbf{z}_i^s,\quad i=1,\dots K$, то стоп,

иначе переход к шагу $s=s+1$.

Для применения алгоритмов кластеризации требуется, как правило, заранее определить из априорных соображений число классов K. Выбор числа классов может сильно повлиять на результат кластеризации, так как неудачный выбор не позволит выделить компактные классы. Поэтому в общем случае возникает необходимость оптимизировать результат кластеризации по числу классов. Заметим, что рассмотренный критерий качества кластеризации не оптимизируется по числу классов, так как имеет максимальное значение (дисперсия исходных данных) при K=1 и минимальное нулевое значение при K=N.

Задача определения оптимального числа классов является весьма сложной теоретической задачей, поэтому часто процедуры кластеризации строятся как алгоритмы, реализующие некоторый способ перебора числа классов. Перебор по числу классов осуществляется, например, так называемыми агломеративными (объединяющими) или дивизимными (разделяющими) процедурами.

Например, в агломеративной иерархической процедуре в самом начале предполагается $K=N$, где N - число объектов, то есть каждый объект является классом. Затем число классов изменяется на $K=N-1$ за счет объединения двух ближайших классов и т.д., вплоть до $K=1$, когда все объекты находятся в одном классе. Затем выбирается оптимальное число классов K^*.

Для объединения классов требуется уточнить понятие двух ближайших классов. Расстояние между двумя классами можно определить различными способами, например:

1. ближайший сосед $\quad d_1\left(\Omega_i,\Omega_j\right)=\min\limits_{\mathbf{x}'\in\Omega_i,\,\mathbf{x}''\in\Omega_j}\left\|\mathbf{x}'-\mathbf{x}''\right\|$;

2. дальний сосед $\quad d_2\left(\Omega_i,\Omega_j\right)=\max\limits_{\mathbf{x}'\in\Omega_i,\,\mathbf{x}''\in\Omega_j}\left\|\mathbf{x}'-\mathbf{x}''\right\|$;

3. среднее расстояние $\quad d_3\left(\Omega_i,\Omega_j\right)=\dfrac{1}{N_iN_j}\sum\limits_{\mathbf{x}'\in\Omega_i}\sum\limits_{\mathbf{x}''\in\Omega_j}\left\|\mathbf{x}'-\mathbf{x}''\right\|$;

4. расстояние по центрам $\quad d_4\left(\Omega_i,\Omega_j\right)=\dfrac{N_iN_j}{N_i+N_j}\left\|\mathbf{z}_i-\mathbf{z}_j\right\|$.

Отметим, что, если классы недостаточно компактны, то результаты могут сильно различаться при использовании различных мер расстояния между классами.

Оптимальное число классов K^* в иерархической процедуре можно оценить по порогу, который задается межклассовыми расстояниями. Предполагается, что ниже оптимального порога происходит частое (естественное) укрупнение классов и быстрое падение их числа при небольших приращениях порога межклассовых расстояний. Выше оптимального порога укруп-

нение классов возобновляется (вынужденно) лишь при относительно большом приращении порога межклассовых расстояний. Процесс объединения можно изобразить в виде дендрограммы, отложив по горизонтали номера объектов, по вертикали - значения расстояний между объединяемыми классами, проводя соединения между ними на соответствующем уровне (Рис. 4.11). Здесь видно, что оптимальное значение порога 0.6 дает оптимальное число классов K^*=3, до которого происходит естественное укрупнение классов.

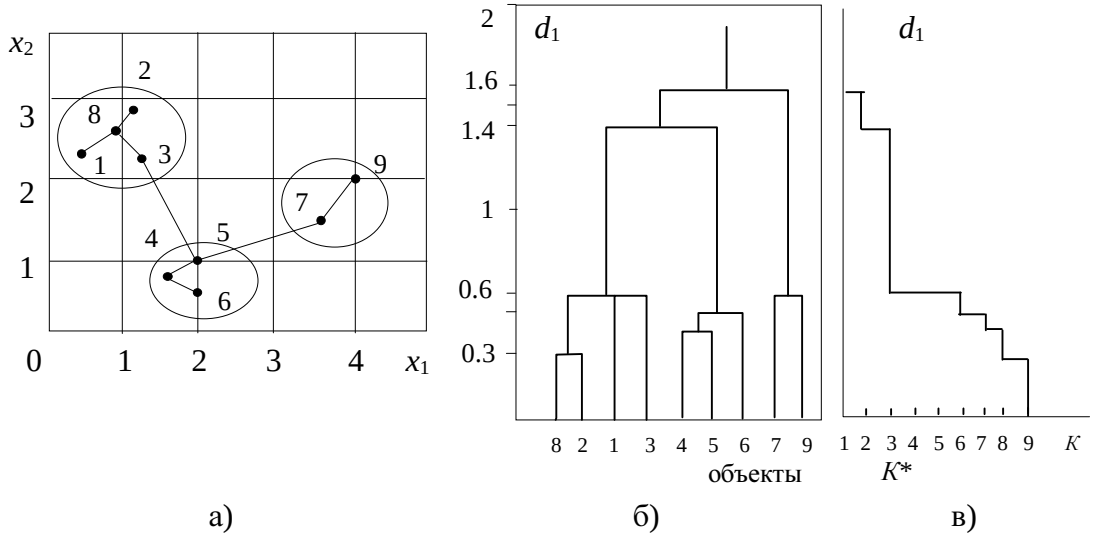


Рис. 4.11. Результат агломеративной иерархической кластеризации

а) граф ближайшего соседства,

б) дендрограмма,

в) изменение порога межклассовых расстояний.

Очевидно, что в дивизимной иерархической процедуре процесс выделения классов выполняется в противоположном направлении. Особенность иерархических группировок состоит в том, что классы разных уровней образуют вложенные подмножества. Такая информация часто важна, так как позволяет исследователю последовательно проследить процесс объединения объектов и уточнить результат объединения. Тем не менее, иерархические алгоритмы вычислительно достаточно трудоемки и требовательны к ресурсам.

Иерархические процедуры группировки реализуют один из способов перебора по числу классов. Альтернативой ему является использование неиерархических алгоритмов, например алгоритма Хартигана. Но тогда при переборе по числу классов придется выполнять кластеризацию заново для каждого значения числа классов K. Алгоритмы такого типа весьма чувствительны к начальному решению. Хорошее начальное решение (исходные центры близки к средним по классам) резко сократит число шагов алгоритма, а плохое (исходные центры являются самыми близкими объектами) может просто привести к неудовлетворительному результату. Так как средние по классам еще только будут найдены, то имеет смысл в качестве исходных центров принять самые далекие объекты. Это оправдано, так как смысл критерия J состоит в том, чтобы сделать классы наиболее компактными, а их центры разнести как можно дальше. Поиск двух самых далеких объектов весьма прост - это максимальный элемент в матрице расстояний. Но поиск $K>2$ самых далеких объектов по матрице расстояний уже является более сложной операцией, трудоемкость которой возрастает с ростом K. Поэтому хотелось бы при переборе по числу классов среди исходных центров для данной кластеризации использовать центры предыдущей кластеризации. Предполагается, что центры классов двух последователь-

ных кластеризаций мало отличаются при больших K. Построим здесь дивизимную неиерархическую процедуру кластеризации.

На каждом шаге, начиная с $K=1$, найдем класс с наибольшей дисперсией, и пусть его номер равен k. Примем в качестве центров два самых далеких объекта в классе k и разобьем его на два класса k и K+1 алгоритмом Хартигана. Для K+1 классов примем в качестве исходных K-1 ранее полученных центров и два новых центра и найдем K+1 классов алгоритмом Хартигана. Перейдем к шагу $K=K+1$. Процедура останавливается при $K=N$.

Оптимальное число классов K^* можно выбрать как границу, до которой среднее дисперсий классов быстро убывает при увеличении K, а после которой уменьшение среднего дисперсий классов резко замедляется. Например, на Рис. 4.12 показано изменение среднего дисперсий классов для данных на Рис. 4.11 а). В данном случае также K^*=3.

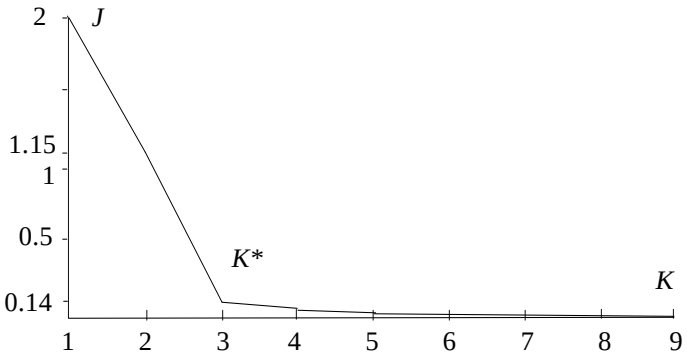


Рис. 4.12. Среднее дисперсий классов.

Если проследить работу данной дивизимной неиерархической процедуры по шагам, то окажется, что последовательность разбиений на классы объектов на Рис. 4.11 а) образует иерархию и отображается дендрограммой Рис. 4.11 б), с той разницей, что дендрограмма строится в противоположном направлении. В общем случае последовательные классификации, полученные данной процедурой, не образуют иерархию. Тем не менее, построенная здесь процедура обладает одним интересным свойством.

Назовем классификацию на K шаге дивизимной неиерархической процедуры устойчивой, если классификация на K+1 шаге иерархически вложена в нее. В противном случае классификацию на K шаге назовем неустойчивой и будем говорить, что на шаге K+1 были нарушения иерархии. Смысл устойчивости классификации ясно следует из свойств процедуры. Действительно, иерархическая вложенность классификации на K+1 класс в классификацию на K классов означает, что класс k был разбит на два, а остальные классы не изменились. Следовательно, две независимые классификации на K шаге процедуры, полученные при разбиении класса k с максимальной дисперсией на два и при разбиении всего множества на K+1 класс, совпали. Это позволяет нам говорить об устойчивости разбиения на K классов и предположить, что были получены компактные классы.

В общем случае в последовательности классификаций можно выделить подпоследовательности иерархически вложенных классификаций, в которых все классификации, кроме последних, являются устойчивыми. Отметим, что классификации на $K=1$ и на $K=N$ классов устойчивы по определению. Каждая подпоследовательность устойчивых классификаций определяет свой масштаб соотношения между между средней дисперсией классов и дисперсией центров. Очевидно, что оптимальное число классов K^* определяется одной из устойчивых классификаций. При выборе оптимального числа классов следует ограничить последователь-

ность устойчивых классификаций справа, отбросив все разбиения с большим числом малонаполненных классов, согласно условию $K^* << N$. Также следует ограничить последовательность устойчивых классификаций слева, отбросив все разбиения, для которых средняя дисперсия классов превышает дисперсию центров. Выполнение данных условий обеспечит получение компактных классов, соответствующих общепринятому в литературе эвристическому определению кластера или сгущения.

Задание на работу

Ознакомиться с теоретической справкой к данной лабораторной работе. Построить алгоритм кластер-анализа. Обработать данные.

Содержание отчета

Номер и название лабораторной работы;
Цель лабораторной работы;
Пояснительная записка к проекту;
Выводы.

Контрольные вопросы

1. Что такое локальное сгущение?
2. Что такое кластер?
3. Опишите различные типы кластеров.
4. Опишите алгоритм k-средних.
5. Опишите модификации алгоритма k-средних (Мак-Куина, Хартигана).

Лабораторная работа №3

ИЗУЧЕНИЕ АЛГОРИТМА ISODADA

Цель и задача работы

Изучить работу алгоритма. Изучить способ решения проблемы выбора числа кластеров.

Теоретические положения

Алгоритм кластеризации ISODATA(ИСОМАД) предназначен для разделения заданного множества образов (в данном случае точек двумерного пространства) на подмножества (кластеры), связанные определенным свойством, например основанное на близости точек по геометрическому расстоянию. Алгоритм эвристический, т.е. результат работы во многом зависит от заданных начальных параметров.

При работе с набором $\{x_1, x_2, ..., x_N\}$, составленным из N элементов, алгоритм ИСОМАД выполняет следующие основные шаги.

Шаг 1. Задаются параметры, определяющие процесс кластеризации:

K—необходимое число кластеров;

Q_N —параметр, с которым сравнивается количество выборочных образов, вошедших в кластер;

Q_s— параметр, характеризующий среднеквадратичное отклонение;

Q_c—параметр, характеризующий компактность;

L— максимальное количество пар центров кластеров, которые можно объединить;

I — допустимое число циклов итерации.

Шаг 2. Заданные N образов распределяются по кластерам, соответствующим выбранным исходным центрам, по правилу

$x \hat{I} S_j$, если $||x - z_j|| < ||x - z_i||$, i=1,2, ..., N_c; i¹ j,

применяемому ко всем образам x, вошедшим в выборку; через S_j обозначено подмножество образов выборки, включенных в кластер с центром z_j.

Шаг 3. Ликвидируются подмножества образов, в состав которых входит менее Q_N элементов, т. е. если для некоторого j выполняется условие $N_j < Q_N$, то подмножество S_j исключается из рассмотрения и значение N_c уменьшается на 1.

Шаг 4. Каждый центр кластера z_j, j=1, 2, ..., N_c, локализуется и корректируется посредством приравнивания его выборочному среднему, найденному по соответствующему подмножеству S_j, т. е.

$$\mathbf{z}_j = \frac{1}{N_j} \sum_{\mathbf{x} \in S_j} \mathbf{x}, \quad j = 1, 2, \ldots, N_c,$$

где N_j —число объектов, вошедших в подмножество Nj.

Шаг 5. Вычисляется среднее расстояние D_j между объектами, входящими в подмножество S_j, и соответствующим центром кластера по формуле

$$\bar{D}_j = \frac{1}{N_j} \sum_{\mathbf{x} \in S_j} \| \mathbf{x} - \mathbf{z}_j \|, \quad j = 1, 2, \ldots, N_c.$$

Шаг 6. Вычисляется обобщенное среднее расстояние между объектами, находящимися в отдельных кластерах, и соответствующими центрами кластеров по формуле

$$\bar{D} = \frac{1}{N} \sum_{j=1}^{N_c} N_j \bar{D}_j.$$

Шаг 7. (а) Если текущий цикл итерации—последний, то задается Q_c=0; переход к шагу 11. (б) Если условие N_c<=*K/2* выполняется, то переход к шагу 8. (в) Если текущий цикл итерации имеет четный порядковый номер или выполняется условие N_c>=*K/2*, то переход к шагу 11; в противном случае процесс итерации продолжается.

Шаг 8. Для каждого подмножества выборочных образов с помощью соотношения

$$\sigma_{ij} = \sqrt{\frac{1}{N_j} \sum_{x \in S_j} (x_{ik} - z_{ij})^2}, \quad i = 1, 2, \ldots, n; \quad j = 1, 2, \ldots, N_c,$$

вычисляется вектор среднеквадратичного отклонения $\mathbf{s}_j = (s_{1j}, s_{2j}, \ldots, s_{nj})'$, где *n* есть размерность образа, x_{ik}, есть i-я компонента *k-ro* объекта в подмножестве S_j, z_{ij} есть i-я компонента вектора, представляющего центр кластера z_j, и N_j — количество выборочных образов, включенных в подмножество S_c. Каждая компонента вектора среднеквадратичного отклонения s_j характеризует среднеквадратичное отклонение образа, входящего в подмножество S_j, по одной из главных осей координат.

Шаг 9. В каждом векторе среднеквадратичного отклонения s_j, j=1, 2, ..., N_c, отыскивается максимальная компонента $s_j$max.

Шаг 10. Если для любого $s_j$max, j=1, 2, ..., N_c, выполняются условия $s_j$max>Q_s,и

а) $\bar{D}_j > \bar{D}$ и $N_j > 2(\theta_N + 1)$

или

б) $N_j < K/2$,

то кластер с центром z_j *расщепляется* на два новых кластера с центрами z_j^+ и z_j^- соответственно, кластер с центром z_j ликвидируется, а значение N_c увеличивается на 1. Для определения центра кластера z_j+ к компоненте вектора z_j, соответствующей максимальной компоненте вектора $\mathbf{s}_j$, прибавляется заданная величина g_j; центр кластера z_j^- определяется вычитанием этой же величины g_j из той же самой компоненты вектора z_j. В качестве величины g_j можно выбрать некоторую долю значения максимальной среднеквадратичной компоненты $s_j$max, т. е. положить $g_j = k s_j$max, где $0 < k <= 1$. При выборе g_j следует руководствоваться в основном тем, чтобы ее величина была достаточно большой для различения разницы в расстояниях от произвольного образа до новых двух центров кластеров, но достаточно малой, чтобы общая структура кластеризации существенно не изменилась.

Если расщепление происходит на этом шаге, надо перейти к шагу 2, в противном случае продолжать выполнение алгоритма.

Шаг 11. Вычисляются расстояния D_{ij} между всеми парами центров кластеров:

D_{ij} =l| z_i - z_j ||, i=1, 2, ..., N_c-*1;* j=i+1, 2, ..., N_c.

Шаг 12. Расстояния D_{ij} сравниваются с параметром Q_c. Те L расстояний, которые оказались меньше Q_c, ранжируются в порядке возрастания:

[$D_{i1j1,}$ $D_{i2j2,}$ ..., $D_{iLjL,}$]

причем $D_{i1j1} < D_{i2j2} < ... < D_{iLjL,}$. *а L*—максимальное число пар центров кластеров, которые можно объединить. Следующий шаг осуществляет процесс слияния кластеров.

Шаг 13. Каждое расстояние D_{iljl} вычислено для определенной пары кластеров с центрами z_{il} и z_{jl}. К этим парам в последовательности, соответствующей увеличению расстояния между центрами, применяется процедура слияния, осуществляемая на основе следующего правила.

Кластеры с центрами z_{il} и z_{jl}, i=1, 2, ..., L, объединяются (при условии, что в текущем цикле итерации процедура слияния не применялась ни к тому, ни к другому кластеру), причем новый центр кластера определяется по формуле

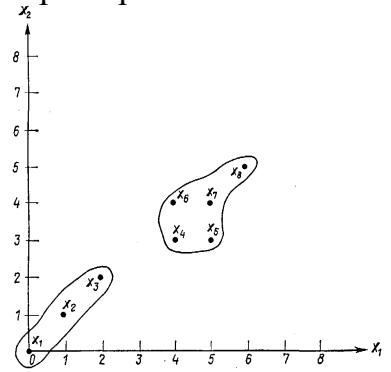
$$\mathbf{z}_l^* = \frac{1}{N_{i_l} + N_{j_l}} \left[N_{i_l}(\mathbf{z}_{i_l}) + N_{j_l}(\mathbf{z}_{j_l}) \right].$$

Центры кластеров z_{il} и z_{jl} ликвидируются и значение N_c уменьшается на 1.

Отметим, что допускается только попарное слияние кластеров и центр полученного в результате кластера рассчитывается, исходя из позиций, занимаемых центрами объединяемых кластеров и взятых с весами, определяемыми количеством выборочных образов в соответствующем кластере. Опыт свидетельствует о том, что использование более сложных процедур объединения кластеров может привести к получению неудовлетворительныхрезультатов. Описанная процедура обеспечивает выбор в качестве центра объединенного кластера точки, представляющей истинное среднее сливаемых подмножеств образов. Важно также иметь в виду, что, поскольку к каждому центру кластера процедуру слияния можно применить только один раз, реализация данного шага ни при каких обстоятельствах не может привести к получению *L* объединенных кластеров.

Шаг 14. Если текущий цикл итерации—последний, то выполнение алгоритма прекращается. В противном случае следует возвратиться либо к шагу 1, если по предписанию пользователя меняется какой-либо из параметров, определяющих процесс кластеризации, либо к шагу 2, если в очередном цикле итерации параметры процесса должны остаться неизменными. Завершением цикла итерации считается каждый переход к шагам I или 2.

Пример.



Выборка образов, использованная для иллюстрации работы алгоритма ИСОМАД.

Хотя алгоритм ИСОМАД не очень подходит для ручных вычислений, принцип его работы можно проиллюстрировать на простом примере. Рассмотрим выборку, образы которой размещены так, как это изображено на рис. 3.11.

В данном случае N ==8 и п = 2. В качестве начальных условий задаем N_c=1, z_1 = (0,0)' и следующие значения параметров процесса кластеризации:

Шаг 1.

К=2, Q_N=1, Q_s=1, Q_c=4, L=0, I=4.

Если всякая *априорная* информация об анализируемых данных отсутствует, эти параметры выбираются произвольным образом и затем корректируются от итерации к итерации.

Шаг 2. Так как задан только один центр кластера, то

$S_1 = \{x_1, x_2, ..., x_8\}$ и $N_1 = 8$.

Шаг 3. Поскольку $N_1 > Q_N$, ни одно подмножество не ликвидируется.

Шаг 4. Корректируется положение центра кластера:

$$\mathbf{z}_1 = \frac{1}{N_1} \sum_{\mathbf{x} \in S_1} \mathbf{x} = \begin{pmatrix} 3{,}38 \\ 2{,}75 \end{pmatrix}.$$

Шаг 5. Вычисляется расстояние D_j:

$$\bar{D}_1 = \frac{1}{N_1} \sum_{\mathbf{x} \in S_1} \| \mathbf{x} - \mathbf{z}_1 \| = 2{,}26.$$

Шаг 6. Вычисляется расстояние *D:*

$D = D_1 = 2{,}26.$

Шаг 7. Поскольку данный цикл итерации—не последний и $N_c = K/2$, осуществляется переход к шагу 8.

Шаг 8. Для подмножества S_1 вычисляется вектор среднеквадратичного отклонения:

$$\boldsymbol{\sigma}_1 = \begin{pmatrix} 1{,}99 \\ 1{,}56 \end{pmatrix}.$$

Шаг 9. Максимальная компонента вектора $\mathbf{s}_1$ равна 1,99, следовательно, $s_1 max = 1{,}99$.

Шаг 10. Поскольку $s_1 max > Q_s$, и $N_c = K/2$, кластер с центром z_1 расщепляется на два новых кластера. Следуя процедуре, предусмотренной шагом 10, выбираем $g_j = 0{,}5 s_j max = 1{,}0$.

Для удобства записи будем называть центры этих кластеров z_1 и z_2 соответственно. Значение N_c увеличивается на 1; переход к шагу 2.

Шаг 2. Подмножества образов имеют теперь следующий вид:

$S_1 = \{x_4, x_5, ..., x_8\}$, $S_2 = \{x_1, x_2, x_3\}$ и $N_1 = 5$, $N_2 = 3$.

Шаг 3. Поскольку обе величины—и N_1, и N_2—больше Q_N, ни одно подмножество не ликвидируется.

Шаг 4. Корректируется положение центров кластеров:

$$\mathbf{z}_1 = \frac{1}{N_1} \sum_{\mathbf{x} \in S_1} \mathbf{x} = \begin{pmatrix} 4{,}80 \\ 3{,}80 \end{pmatrix}, \quad \mathbf{z}_2 = \frac{1}{N_2} \sum_{\mathbf{x} \in S_2} \mathbf{x} = \begin{pmatrix} 1{,}00 \\ 1{,}00 \end{pmatrix}.$$

Шаг 5. Вычисляется расстояние D_j, j=1,2:

$$\bar{D}_1 = \frac{1}{N_1} \sum_{\mathbf{x} \in S_1} \| \mathbf{x} - \mathbf{z}_1 \| = 0{,}80,$$

$$\bar{D}_2 = \frac{1}{N_2} \sum_{\mathbf{x} \in S_2} \| \mathbf{x} - \mathbf{z}_2 \| = 0{,}94.$$

Шаг 6. Вычисляется расстояние *D:*

$$\bar{D} = \frac{1}{N} \sum_{j=1}^{N_c} N_j \bar{D}_j = \frac{1}{8} \sum_{j=1}^{2} N_j \bar{D}_j = 0{,}85.$$

Шаг 7. Поскольку данная итерация имеет четный порядковый номер, условие (в) шага 7 выполняется. Поэтому следует перейти к шагу 11.

Шаг 11. Вычисление расстояний между парами центров кластеров:

$$D_{12} = \| \mathbf{z}_1 - \mathbf{z}_2 \| = 4{,}72.$$

Шаг 12. Величина расстояния D_{12} сопоставляется с параметром Q_c. В данном случае $D_{12} > Q_c$.

Шаг 13. Результаты шага 12 показывают, что объединение кластеров невозможно.

Шаг 14. Поскольку данный цикл итерации—не последний, необходимо принять решение: вносить или не вносить изменения в параметры процесса кластеризации. Так как в данном (простом) случае 1) число выделенных кластеров соответствует заданному, 2) расстояние между ними больше среднего разброса, характеризуемого среднеквадратичными отклонениями, и 3) каждый кластер содержит существенную часть общего количества выборочных образов, то делается вывод о том, что локализация центров кластеров правильно отражает специфику анализируемых данных. Следовательно, переходим к шагу 2.

Шаги 2—6 дают те же результаты, что и в предыдущем^ цикле итерации.

Шаг 7. Ни одно из условий, проверяемых при реализации данного шага, не выполняется. Поэтому переходим к шагу 8.

Шаг 8. Для множеств $S_1 = \{x_4, x_5, ..., x_8\}$, $S_2 = \{x_1, x_2, x_3\}$

$$\sigma_1 = \begin{pmatrix} 0{,}75 \\ 0{,}75 \end{pmatrix}, \quad \sigma_2 = \begin{pmatrix} 0{,}82 \\ 0{,}82 \end{pmatrix}.$$

Шаг 9. В данном случае $s_1 \max = 0{,}75$ и $s_2 \max = 0{,}82$.

Шаг 10. Условия расщепления кластеров не выполняются. Следовательно, переходим к шагу 11. .

Шаг 11. Полученный результат идентичен результату последнего цикла итерации

$$D_{12} = \| \mathbf{z}_1 - \mathbf{z}_2 \| = 4{,}72.$$

Шаг 12. Полученный результат идентичен результату последнего цикла итерации.

Шаг 13. Полученный результат идентичен результату последнего цикла итерации.

Шаг 14. На данном цикле итерации не были получены новые-результаты, за исключением изменения векторов среднеквадратичного отклонения. Поэтому переходим к шагу 2.

Шаги 2—6 дают те же результаты, что и в предыдущем;

цикле итерации.

Шаг 7. Поскольку данный цикл итерации—последний, задаем $Q_c = 0$ и переходим к шагу 11.

Шаг 11. Как и раньше,

$$D_{12} = \| \mathbf{z}_1 - \mathbf{z}_2 \| = 4{,}72.$$

Шаг 12. Полученный результат идентичен результату последнего цикла итерации.

Шаг 13. Результаты шага 12 показывают, что объединение кластеров невозможно.

Шаг 14. Поскольку данный цикл итерации—последний, выполнение алгоритма заканчивается.

Даже из этого простого примера должно быть ясно, что применение алгоритма ИСОМАД к набору данных умеренной сложности в принципе позволяет получить интересные результаты только после проведения обширных экспериментов. Выявление структуры данных может быть, однако, существенно ускорено благодаря эффективному использованию информации, получаемой после каждого цикла итерационного процесса. Эту информацию, как будет показано ниже, можно использовать для коррекции параметров процесса кластеризации непосредственно при реализации алгоритма.

Задание на работу

Ознакомиться с теоретической справкой к данной лабораторной работе. Реализовать алгоритм. Определить оптимальное число кластеров.

Содержание отчета

Номер и название лабораторной работы;
Цель лабораторной работы;
Пояснительная записка к проекту;
Выводы.

Контрольные вопросы

1. Что такое локальное сгущение?
2. Что такое кластер?
3. Опишите различные типы кластеров.
4. Опишите алгоритм k-средних.
5. Опишите модификации алгоритма k-средних (Мак-Куина, Хартигана).
6. В чем суть проблемы выбора числа классов?
7. Что такое дендрограмма?
8. Постройте дивизимный алгоритм классификации.
9. Постройте агломеративный алгоритм классификации.
10. Опишите дивизимный неиерархический алгоритм классификации.
13. Опишите алгоритм ISODATA.

Лабораторная работа №4

ИЗУЧЕНИЕ АЛГОРИТМА FOREL

Цель и задача работы

Изучить работу алгоритма. Изучить способ решения проблемы выбора числа кластеров.

Теоретические положения

АЛГОРИТМЫ ФОРЕЛЬ И ФОРЕЛЬ 2

Алгоритм Форель является примером эвристического дивизимного алгоритма классификации. В основе работы алгоритма Форель лежит использование **гипотезы компактности**: близким в содержательном смысле объектам в геометрическом пространстве признаков соответствуют обособленные множества точек, так называемые «сгустки». Если расстояние между центром n-го таксона и точкой kэтого таксона обозначить s_{nk}, то сумма расстояний между центром и всеми точками kэтого таксона будет равна:

$$P_n = \sum_{k=1}^{L} s_{nk}$$

где:

P_n – расстояние между центром n-го таксона и всеми точками этого таксона;

- s_{nk} – расстояние между центром n-го таксона и точкой kэтого таксона.

Сумма таких внутренних расстояний для всех nтаксонов равна:

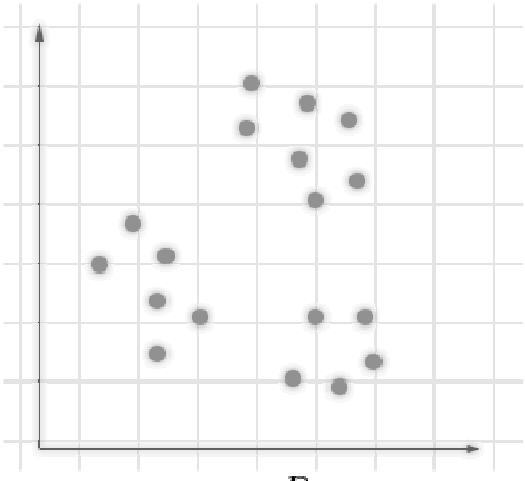
$$P = \sum_{n=1}^{N} P_n$$

Целью работы алгоритма Форель является найти такое разбиение множества объектов на n таксонов, чтобы величина Pбыла минимальной.

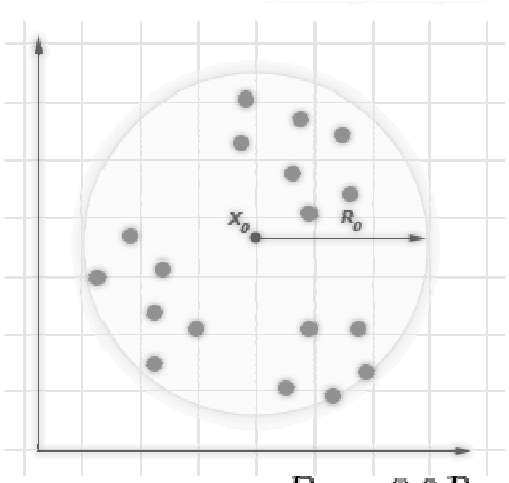
Работа алгоритма заключается в перемещении гиперсферы определенного радиуса в геометрическом пространстве до получения устойчивого центра тяжести наблюдений, попавших в эту гиперсферу. До начала работы алгоритма признаки объектов нормируются так, чтобы их значения находились между нулем и единицей

Пример работы алгоритма.

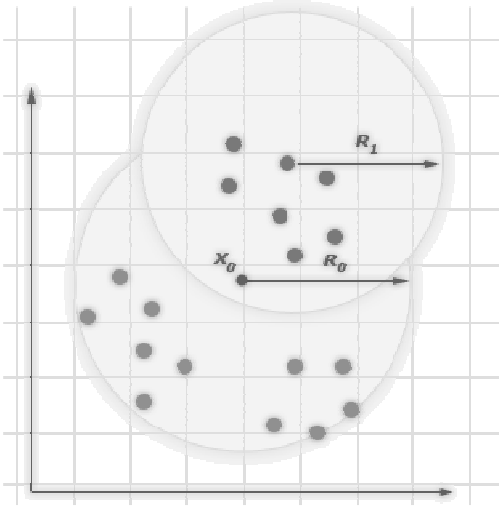
Допустим, было дано некоторое множество классифицируемых объектов. Пусть каждый объект обладает только двумя свойствами; это позволит отобразить исходные данные на геометрической плоскости:



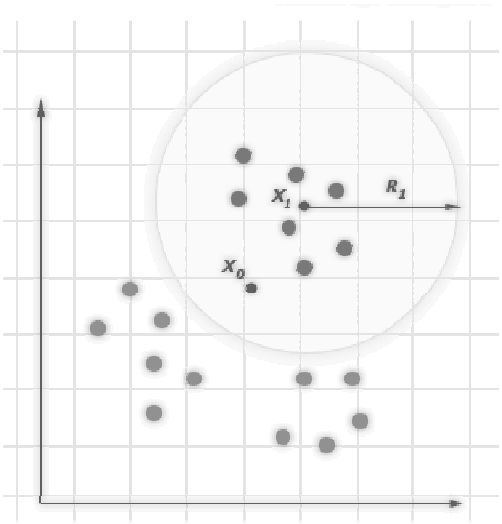
Шаг 1. Построить гиперсферу радиуса R_0 охватывающую все множество точек:



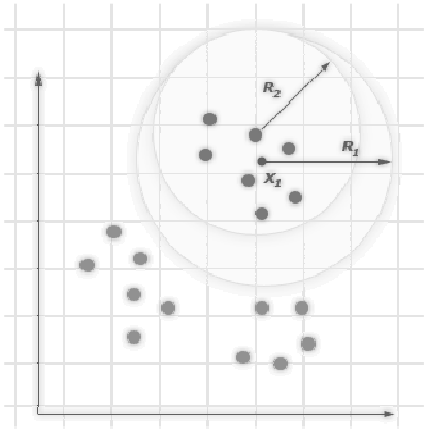
Шаг 2. Установить радиус гиперсферы $R_1 = 0.9R_0$ и перенести центр сферы в любую из внутренних точек (расстояние до которых меньше радиуса):



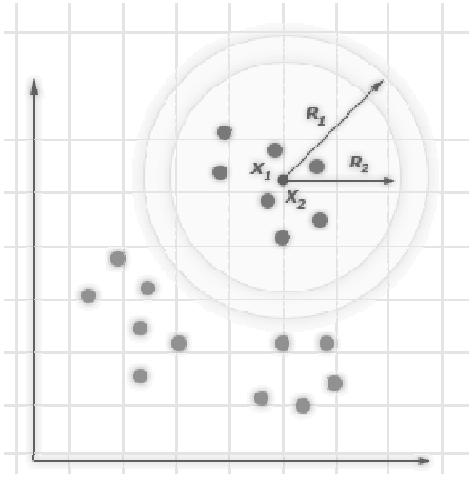
Шаг 3. Вычислить новый центр тяжести и перенести в него центр сферы:



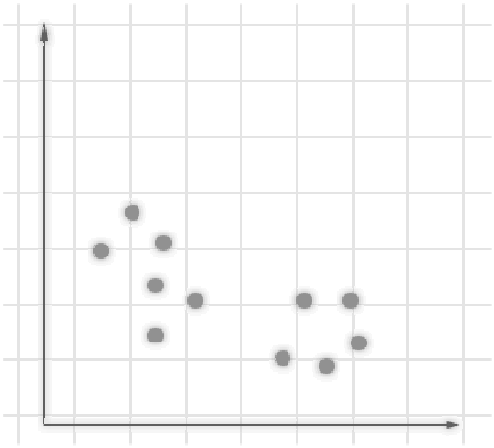
Шаг 4. Если новый центр тяжести отличается от предыдущего необходимо вернуться к шагу 2 и повторить цикл. Цикл будет повторяться до тех пор пока центр тяжести не перестанет смещаться. Таким образом, центр сферы перемещается в область локального сгущения точек. В предложенном примере центр сферы $X_0 \neq X_1$, поэтому: необходимо установить новый радиус сферы $R_2 = 0.9R_1$ и перенести центр сферы в произвольную внутреннюю точку:



Шаг 5. Вычислить новый центр тяжести и перенести в него центр сферы. Новый центр тяжести $X_2 = X_1$, поэтому внутренние точки текущей сферы объединяются в таксон:



Шаг 6. Точки принадлежащие новому таксону исключаются из анализа и работа алгоритма повторяется с шага №1. И так до тех пор пока все точки не будут исключены из анализа:



Процедура алгоритма Форель является сходящейся за конечное число шагов в евклидовом пространстве любой размерности при произвольном расположении точек и любом выборе гиперсферы.

Если начальную точку, в которую переносится центр сферы, на шаге №2 менять случайным образом, может получиться несколько вариантов таксономии, из которых выбирается тот, на котором достигается $MIN(P)$.

Алгоритм Форель 2 является модификацией исходного алгоритма и применяется в тех случаях, когда необходимо получить изначально заданное количество кластеров (таксонов). Радиус сферы по мере надобности может изменяться на заданную величину, которая от итерации к итерации будет уменьшаться.

Наилучшему варианту таксономии отвечает $MIN(P)$ при числе таксонов равном заданному.

Задание на работу

Ознакомиться с теоретической справкой к данной лабораторной работе. Построить алгоритм кластер-анализа. Обработать данные.

Содержание отчета

Номер и название лабораторной работы;
Цель лабораторной работы;
Пояснительная записка к проекту;
Выводы.

Контрольные вопросы

1. Что такое локальное сгущение?
2. Что такое кластер?
3. Опишите различные типы кластеров.
4. Опишите варианты алгоритма Forel.

Лабораторная работа №5

ИЗУЧЕНИЕ АЛГОРИТМОВ SKAT, KOLAPS

Цель и задача работы

Изучить работу алгоритмов. Изучить способ решения проблемы выбора числа кластеров.

Теоретические положения

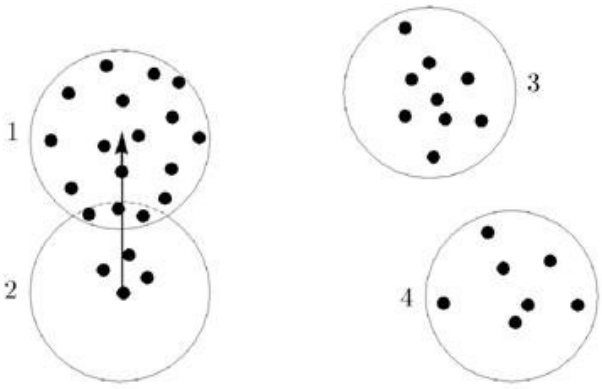
Алгоритм SKAT

Если при многократном случайном выборе начальной точки получается большое число неодинаковых таксономий или если таксоны сильно отличаются друг от друга по количеству своих точек, то это может означать, что наш материал наряду с несколькими локальными сгустками точек содержит еще и одиночные точки или небольшие их скопления, случайно разбросанные в пространстве между сгустками. Создается ощущение того, что имеется несколько «самостоятельных» таксонов и ряд случайно образовавшихся, «несамостоятельных» таксонов, которые было бы целесообразно присоединить к ближайшим самостоятельным.

Каждый очередной таксон находился нами в условиях, когда точки, попавшие в предыдущие таксоны, исключались из рассмотрения. А что происходит, если таксоны формируются в присутствии всех m точек? Может случиться, что некоторые из более поздних таксонов включат в свой состав точки, ранее вошедшие в другие таксоны, и не останутся на месте, а станут скатываться в сторону соседнего сгустка точек и сольются с одним из своих предшественников. Такая ситуация изображена на рис. 6: таксон 2 начнет смещаться и сольется с таксоном 1. Другие же таксоны останутся на прежнем месте и с прежним составом своих внутренних точек. Будем считать таксоны 1, 3 и 4 устойчивыми, самостоятельными, а таксон 2 — неустойчивым, случайным. Случайные таксоны могут появляться из-за помех в данных или из-за неудачного выбора радиуса сфер.

Проверку на устойчивость таксономии можно было бы делать строгими статистическими методами. Однако они разработаны для случаев, когда речь идет о простых распределениях (обычно нормальных) в пространстве малой размерности. Анализ данных же часто имеет дело с относительно небольшим числом объектов (прецедентов) в пространстве большой размерности, и говорить о каком бы то ни было распределении не возможно. Поэтому приходится применять единственно возможные в такой ситуации и, как кажется, достаточно разумные эвристические приемы.

Один из таких приемов реализован в алгоритме SKAT. На вход программы подается множество m объектов и результаты его таксономии S с помощью алгоритма FOREL при радиусе сферы, равном R. Процедуры таксономии повторяются с таким же радиусом сфер, но теперь в качестве начальных точек выбираются центры, полученные в таксономии S, и формирование каждого нового таксона делается с участием всех m точек. В результате обнаруживаются неустойчивые таксоны, которые скатываются к таксонам-предшественникам.

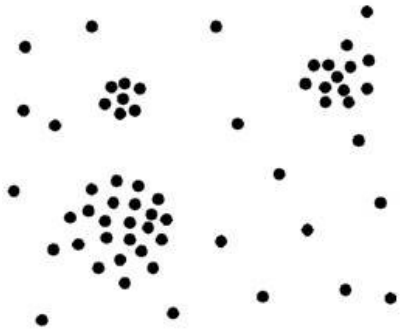


Решение выдается в виде перечня устойчивых таксонов и указания тех неустойчивых, которые к ним тяготеют. Если мы хотим ограничиться только устойчивыми таксонами, тогда мы должны стремиться к такому варианту таксономии, при котором количество точек в устойчивых таксонах было бы максимальным, т. е. максимизировать функционал

$$F = \sum_{j=1}^{k} m_j f(j)$$, где $$f(j) = \begin{cases} 1, \text{ если } j-\text{й таксон устойчив,} \\ 0, \text{ если } j-\text{й таксон неустойчив.} \end{cases}$$

Алгоритм KOLAPS

Взглянув на рис., можно отметить следующие его особенности: здесь выделяются три разных по диаметру сгустка точек на равномерном сером фоне. Хотелось бы, чтобы алгоритм таксономии мог выделить эти три сгустка, каждый со своим диаметром, отделив их от точек фона, т. е. мог бы решать задачу выделения ярких созвездий на звездном небе. Для решения таких задач предназначен один из алгоритмов семейства FOREL — алгоритм KOLAPS. Его можно разделить на два этапа.



На первом этапе ищутся потенциальные центры будущих таксонов, а на втором — делается проверка, действительно ли выбранная точка является центром устойчивого таксона.

В начале первого этапа сфера достаточно большого радиуса $R < R_0$ помещается в любую точку множества и смещается, как и в алгоритме FOREL, в центр локального сгустка точек. Количество m_j внутренних точек полученного таксона служит мерой локальной плотности точек в данном месте признакового пространства. Если m_j больше некоторого порога d, то центр такого мощного таксона за-

носится в список претендентов на роль центра таксона-созвездия, а попавшие в него внутренние точки из дальнейшего рассмотрения исключаются. Если m_j меньше d, то список претендентов не меняется, но внутренние точки этого таксона также гасятся. Затем центр сферы помещается в любую из оставшихся точек и процесс выделения следующих таксонов продолжается до исчерпания всех точек.

После этого восстанавливается все множество m точек и выделяется в списке претендентов таксон с наибольшим значением локальной плотности m_j. В центр этого таксона помещается сфера, и ее радиус начинает сжиматься от величины R до величины $R_{\min}$. На каждом i-м шаге сжатия определяется число точек, оставшихся внутренними. Если начальный радиус был слишком большим для данного таксона (т. е. если он захватывал много разреженного пространства), то в начале процесса сжатия скорость убывания числа внутренних точек будет небольшой. По мере вхождения сферы в более плотную часть таксона количество теряемых точек начнет увеличиваться, что служит сигналом к остановке сжатия. В результате находится наиболее естественный для данного таксона радиус сферы и фиксируется число его внутренних точек m'_j. Та же процедура постепенного сжатия повторяется и для других таксонов из списка претендентов, упорядоченных по характеристике локальной плотности. В результате выбирается k таких таксонов, в состав которых после сжатия попадает наибольшее количество точек. Это условие равнозначно максимизации функционала $F = \sum_{j=1}^{k} m'_j$.

Задание на работу

Ознакомиться с теоретической справкой к данной лабораторной работе. Построить алгоритмs кластер-анализа. Обработать данные.

Содержание отчета

Номер и название лабораторной работы;
Цель лабораторной работы;
Пояснительная записка к проекту;
Выводы.

Контрольные вопросы

1. Что такое локальное сгущение?
2. Что такое кластер?
3. Опишите различные типы кластеров.
4. Опишите алгоритмы Skat, Kolaps.