1.3. Данные в виде картинки

1.3.1. Задача визуализации данных

В этом разделе нашей целью является дать обзор тех методов, которые в настоящее время используются для визуального представления сразу всей структуры многомерного набора данных. Для визуализации могут быть использованы 1-, 2- и 3-мерные пространства отображений, но мы в своем рассмотрении практически целиком ограничимся способом визуализации с помощью 2-мерных поверхностей, поскольку именно в таком виде человек воспринимает геометрические структуры наиболее естественно и отношения между объектами выглядят наиболее наглядно.

Под ***визуализацией данных*** мы понимаем такой ***способ представления многомерного распределения данных на двумерной плоскости, при котором, по крайней мере, качественно отражены основные закономерности, присущие исходному распределению – его кластерная структура, топологические особенности, внутренние зависимости между признаками, информация о расположении данных в исходном пространстве*** и т.д. В качестве основных применений методов визуализации можно указать следующие:

а) наглядное представление геометрической метафоры данных;

б) лаконичное описание внутренних закономерностей, заключенных в наборе данных;

в) сжатие информации, заключенной в данных;

г) восстановление пробелов в данных;

д) решение задач прогноза и построения регрессионных зависимостей между признаками.

Мы кратко рассмотрим традиционные методы, решающие поставленную задачу непосредственным и несколько громоздким образом – это *целенаправленное проецирование* данных и *многомерное шкалирование*. Затем обратимся к очень популярному в последнее время способу визуализации с помощью самоорганизующихся карт Кохонена. Глава будет завершена рассмотрением нового подхода, разработанного группой «Нейрокомп» на базе Института Вычислительного Моделирования г.Красноярска и названного **методом упругих карт**.

1.3.2. Методы целенаправленного проецирования в пространства малой размерности

Один из способов поставить задачу представления данных в виде двумерной картинки заключается в следующем: найти такое отображение (способ проецирования) из исходного пространства на двумерную плос-

кость, которое бы оптимизировало заданный критерий качества – некоторый функционал от координат точек данных до и после процедуры проецирования. Такая постановка задачи лежит в основе совокупности подходов, объединяемых под названием *целенаправленное проецирование* (в зарубежной литературе – projecting pursuit) в пространство малой размерности.

Можно выделить два варианта решения этой задачи:

1. Вид отображения U известен заранее и является, как правило, линейным отображением на плоскость. Оптимизируемый функционал в данном случае называется *проекционным индексом* и обозначается $Q(U,X)$, под X понимается весь набор многомерных данных, Q зависит от параметров отображения. В зависимости от поставленной задачи могут быть использованы следующие проекционные индексы:

а) индекс, минимизирующий расстояние от точек данных до их проекций – и это дает классический метод снижения размерности с помощью главных компонент;
б) индексы, максимизирующие расстояния между кластерами (один из вариантов таких индексов максимизирует энтропию конечного двумерного распределения данных);
в) индексы, максимально разделяющие заранее заданные классы для построения линейного классификатора;
г) индексы, использующиеся для выделения *аномальных наблюдений*, далеко отстоящих от основной массы распределения точек данных;
д) индексы, выделяющие нелинейные структуры в многомерных данных.

Явный вид этих проекционных индексов приведен в литературе по прикладной статистике [2].

2. Вид отображения заранее неизвестен. Тогда оптимизируемый критерий является функцией от набора двумерных координат, приписанной каждой точке данных. Задачей в этом случае является назначить каждой из точек исходного набора данных пару координат таким образом, чтобы минимизировать функционал, описывающий «меру искажения» структуры данных.

Одним из самых популярных является функционал, являющийся аналогом стресса в многомерном шкалировании и описывающий меру искажения взаимных расстояний между точками в исходном и результирующем пространстве отображения.

Остановимся здесь на одном важном для нашего изложения моменте. В разделе, посвященном квазилинейным моделям мы уже сталкивались с ситуацией, когда каждой точке данных можно было бы приписать две координаты (метод главной плоскости). Это позволяет построить в пространстве данных гладкое многообразие, которое обладает свойством обобщать заключенную в данных информацию и служить для лаконичного описания, сжатия информации или для восстановления пробелов в данных. Совершенно аналогично над методом проецирования в пространство меньшей размерности можно надстроить процедуру построения моделирующей двумерной поверхности, вложенной в многомерное пространство признаков.

1.3.3. Многомерное шкалирование

Иногда исходная информация бывает изначально представлена не в виде таблицы типа «объект-признак», а в виде квадратной таблицы удаленностей объектов друг от друга. На пересечении i-ой строки и j-ого столбца в такой таблице стоит оценка расстояний от i-го до j-го объекта. Такой вид представления информации характерен для психологических исследований, когда человеку предлагается оценивать сходство или различие в некоторой системе объектов или понятий.

Таким образом, изначально каждому объекту не сопоставляется никакой координаты в многомерном пространстве и представить такую информацию в виде геометрической метафоры затруднительно. Задача *многомерного шкалирования* заключается в том, чтобы сконструировать распределение данных в пространстве таким образом, чтобы расстояния между объектами в соответствовали исходно заданным в матрице удаленностей. Возникающие координатные оси могут быть интерпретированы как некоторые неявные факторы, значения которых определяют различия объектов между собой. Если попытаться сопоставить каждому объекту пару координат, то в результате мы получим способ визуализации данных.

Различают два основных алгоритма многомерного шкалирования – метрический и неметрический, хотя сами вычислительные процедуры этих алгоритмов практически не отличаются [4,14,43].

1. В основе *метрического многомерного шкалирования* лежит допущение о том, что расстояния в таблице удаленностей соответствуют реальным расстояниям между объектами в конструируемом пространстве признаков.

В линейном методе метрического шкалирования применяется метод главных компонент, но не к исходной матрице расстояний, а к так называемой *дважды центрированной матрице*, в которой среднее значение чисел в любой строке и столбце равно нулю. Дважды центрированная матрица однозначно вычисляется по исходной. После этого существует возможность определить размерность пространства, обеспечивающего *точное*

воспроизведение матрицы удаленностей, либо определить эффективную размерность конструируемого пространства признаков, которая обеспечит воспроизведение матрицы удаленностей с заданной точностью.

В нелинейных методах размерность пространства задается изначально и с помощью градиентных методов оптимизируется функционал качества, описывающий меру искажения матрицы удаленностей. Этот функционал, называемый *стрессом*, уже упоминался нами в предыдущем разделе и мы вернемся к нему позже.

2. В *неметрическом многомерном шкалировании* предполагается, что удаленность объектов измерена в ординальной шкале, то есть важны не столько сами численные значения попарных расстояний, сколько их ранговый порядок. Процедуры неметрического шкалирования строят такую геометрическую конфигурацию точек в q-мерном пространстве, чтобы ранговые порядки расстояний совпали, по возможности, с ранговыми порядками исходных расстояний. Для оценки качества выбранных ранговых координат применяется все тот же критерий стресса.

Аналогично традиционному факторному анализу, в многомерном шкалировании существует неоднозначность выбора координат, связанная с тем, что координатную систему в полученном пространстве можно произвольным образом повернуть – расстояния между объектами при этом не изменяются. Как правило, поворот осуществляют таким образом, чтобы либо полученные координатные оси имели максимально наглядную интерпретацию, либо значения определенных признаков оказались максимально скоррелированы.

1.3.4. Вложенные поверхности

Итак, в основе методов целенаправленного проецирования и многомерного шкалирования лежит идея оптимизации некоторого функционала, который зависит от начального положения точек в пространстве и конечного расположения точек на двумерной плоскости. Выбирая различные виды функционалов, можно строить различные проекции данных, на которых будут подчеркнуты те или иные их особенности. В целом такой подход является достаточно прозрачным и ясным, но при его практическом использовании возникают определенные трудности.

Во-первых, задача оптимизации нелинейной функции является трудной сама по себе. В упомянутых методах используются, как правило, градиентные процедуры, требующие больших вычислительных затрат, которые растут пропорционально квадрату от числа точек данных (нужно вычислять все попарные расстояния в пространстве отображений). Это делает весьма затруднительной практическую реализацию этих алгоритмов для таблиц данных, содержащих большое (порядка нескольких тысяч) число строк.

Во-вторых, оказывается, что выразительная картина многомерного распределения данных, изображенная на двумерной картинке еще не решает всех вопросов, которые может поставить себе исследователь. Заманчива идея наносить на двумерную карту не только сами точки данных, но и разнообразную информацию, сопутствующую данным – например, отображать так или иначе положение точек в исходном пространстве, плотности различных подмножеств, другие непрерывно распределенные величины, заданные в исходном пространстве признаков. Все это подталкивает к мысли использовать как можно полнее тот «фон», на который наносятся данные, а также вид самих точек данных для отображение различной количественной и аттрибутивной информации.

Наконец, после того, как данные нанесены на двумерную плоскость, хотелось бы, чтобы появилась возможность расположить на двумерной плоскости те данные, которые не участвовали в настройке отображения. Это позволило бы, с одной стороны, использовать полученную картину для построения различного рода экспертных систем и решать задачи распознавания образов, с другой – использовать ее для восстановления данных с пробелами.

Таким образом можно подойти к идее использования для визуализации данных и извлечения информации некоторого вспомогательного объекта, который в дальнейшем мы будем называть *картой*. Этот объект представляет из себя ограниченное двумерное нелинейное многообразие, вложенное в многомерное пространство данных таким образом, чтобы служить моделью данных, то есть форма и расположение такого многообразия должна отражать основные особенности распределения множества точек данных.

Простой пример карты данных – плоскость первых двух главных компонент. Как мы уже упоминали, среди всех двумерных плоскостей, вложенных в пространство она служит оптимальным экраном, на котором можно отобразить основные закономерности, присущие данным. В качестве другой, еще более простой (но не оптимальной) карты можно использовать любую координатную плоскость любых двух выбранных координат.

> ✂ Среди различных проекций на пары координатных осей наиболее информативными будут те, где в качестве координат выбираются наиболее значимые признаки, например, те, которые имеют наибольший вес в векторе, задающем направление первой главной компоненты. ✂

Обобщением способа представлять данные с помощью метода главных компонент будет случай, когда карта может иметь любую нелинейную форму, не используя в процессе построения карты никаких гипотез о распределении данных. На пути создания такой нелинейной модели данных необходимо ответить на следующие вопросы:

1. Как описывать расположение карты в пространстве?

Для того, чтобы описывать в многомерном пространстве вложенное двумерное многообразие, используют обычно вектор-функцию $\boldsymbol{r} = \boldsymbol{r}(u,v)$ от двух координат u, v, которые называются *внутренними* координатами или параметрами. Линии, вдоль которых одна из внутренних координат принимает постоянное значение, задают на поверхности внутреннюю координатную сетку. Таким образом, любая точка на поверхности задается, с одной стороны, только двумя внутренними координатами (именно поэтому размерность многообразия, задаваемого формулой $\boldsymbol{r} = \boldsymbol{r}(u,v)$ равна по построению двум), а с другой стороны, будучи точкой в m-мерном пространстве имеет m значений координат в исходном пространстве.

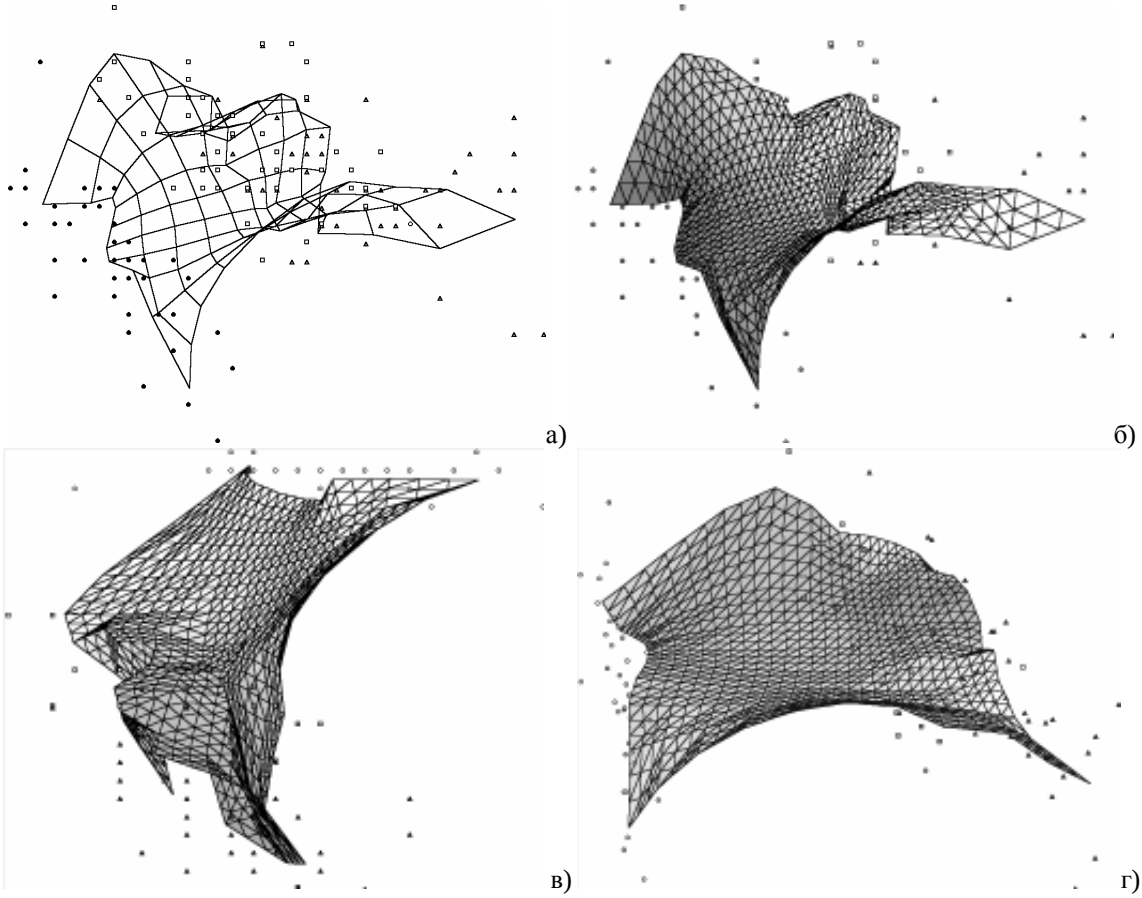


Рис. 15. Вид построенной карты с точки зрения различных двумерных плоскостей-экранов.
На проекциях показаны точки данных, соответствующие упоминавшейся базе данных по цветкам ириса.
а) вид построенной сетки на координатной плоскости «длина лепестка – ширина лепестка»;
б) вид карты, для которой применена процедура кусочно-линейной интерполяции между узлами;
в) вид карты в координатах «ширина лепестка – ширина чашелистика»;

Для вычислительных процедур гораздо удобнее производить операции не с самим многообразием, а с его точечной аппроксимацией, задаваемой с помощью сетки узлов (удобно, если в этих узлах значения внутренних координат принимают целые значения). Для описания положения прямоугольной сетки узлов в пространстве достаточно $m \cdot p \cdot q$ чисел, где m – размерность пространства, а p и q – число узлов прямоугольной сетки по вертикали и горизонтали. Если число узлов сетки гораздо меньше числа точек данных, то используя такую сетку в качестве модели данных, можно получить сжатие информации, заключенной в данных, с точностью, зависящей от способа построения карты и особенностями структуры данных.

Изначально карта может быть задана с помощью плоской равномерной сетки узлов, как-то размещенных в пространстве признаков. Под действием тех или иных вычислительных процедур карта может искривляться, прилегая к данным и отражая особенности их структуры.

После того, как получена точечная аппроксимация многообразия, для того, чтобы восстановить карту нужно воспользоваться подходящей процедурой интерполяции между узлами. Самым простым вариантом интерполяции является кусочно-линейная интерполяция. Для ее построения изначально на сетке реализуется тот или иной вариант триангуляции, в результате чего карта состоит из отдельных треугольных кусков плоскостей.

На рис.15 показано, как может выглядеть построенная карта с точки зрения различных плоских двумерных экранов, расположенных в пространстве – разных координатных плоскостей и плоскости главных компонент.

2. Каким образом сопоставить каждой точке данных точку на карте?

После того, как многообразие построено, для визуализации данных необходимо указать правило, с помощью которого данные из исходного пространства признаков переносятся на карту. Предполагается, что длина вектора переноса не будет слишком велика, поскольку карта аппроксимирует данные и достаточно плотно в среднем к ним прилегает.

Простейшим способом переноса или *проецирования* является сопоставление каждой точке данных ближайшего узла сетки. Такой способ даже не требует доопределения сетки до многообразия и разбивает все множество данных на $p \times q$ подмножеств – *таксонов*, внутри каждого из них ближайшим является один и тот же узел карты. В некоторых задачах такой способ проектирования является приемлемым, однако, он не дает непрерывного отображения пространства данных на двумерное многообразие – при переходе от узла к узлу функция отображения имеет разрыв – и поэтому такое проектирование называется *кусочно-постоянным*.

Другой идеей, которая может быть применена при проецировании, является сопоставление точке данных *ближайшей точки* на карте (а не ближайшего узла!). В случае гладкого многообразия нахождение такой

точки может быть связано с определенными вычислительными трудностями, в случае же упомянутой линейной интерполяции между узлами достаточно просто ближайшую точку многообразия определить достаточно просто. Соответствующий алгоритм приведен в разделе 2.5.6, а здесь укажем, что ближайшей точкой карты может оказаться точка внутри треугольного куска плоскости, образующего *грань* карты или точка внутри отрезка, соединяющего два соседних узла, образующего *ребро* карты, а также ближайшей точкой может оказаться узел (*вершина*) карты. Соответственно, в случае кусочно-линейного многообразия пространство вокруг карты разбивается на области, для которых ближайшей является узел, ребро или грань построенной карты. Такой способ проецирования оказывается кусочно-линейным.

Для проецирования многомерных данных на двумерную кусочно-линейную поверхность могут применяться другие способы кусочно-линейного проецирования, например, центральное проецирование. Центр проекции определяется с помощью координат ближайшего узла сетки и прилегающих к нему соседей, то есть для каждого таксона данных будет найден свой центр проекции.

Возможно построение кусочно-гладкого проектора с помощью построения в каждом узле карты некоторого двумерная гладкая поверхность, аппроксимирующая в окрестности узла карту. Наиболее прост случай построения поверхности второго порядка, однако, в этом случае не удается сшить построенные куски поверхностей и в результате построенное многообразие будет выглядеть как «папье-маше» – на каждый из узлов приклеивается поверхность второго порядка с вершиной или седловой точкой в узле.

На рис.16 сравниваются два способа проецирования – в ближайший узел и в ближайшую точку карты. На нем же показана карта в своих внутренних координатах (развернутая на плоскости) и расположение точек, полученных в результате этих способов проецирования.

Итак, на построенной карте можно разместить точки данных. Кроме этого, карта сама по себе обладает рядом свойств: прилегая к данным, карта в одних областях пространства сжимается, в других растягивается, и это задает на карте двумерную метрику, разные точки карты имеют разные координаты в пространстве, карта по-разному прилегает к данным в разных областях и т.д. Таким образом, карта сама по себе служит носителем разнообразной информации.

Теперь мы обратимся к конкретным способам построения моделирующих карт.

1.4. Самоорганизующиеся карты Кохонена и их приложения

Тойво Кохонен предложил нейросетевую архитектуру для автоматической кластеризации (классификации без учителя), в которой учитывается информация о взаимном расположении нейронов, которые образуют решетку. Сигнал в такую нейросеть поступает сразу на все нейроны, а веса соответствующих синапсов интерпретируются как координаты положения узла и выходной сигнал формируется по принципу «победитель забирает все» - то есть ненулевой выходной сигнал имеет нейрон, ближайший (в смысле весов синапсов) к подаваемому на вход объекту.. В процессе обучения веса синапсов настраиваются таким образом, чтобы узлы решетки «располагались» в местах локальных сгущений данных, то есть описывали кластерную структуру облака данных, с другой стороны, связи между нейронами соответствуют отношениям соседства между соответствующими кластерами в пространстве признаков.

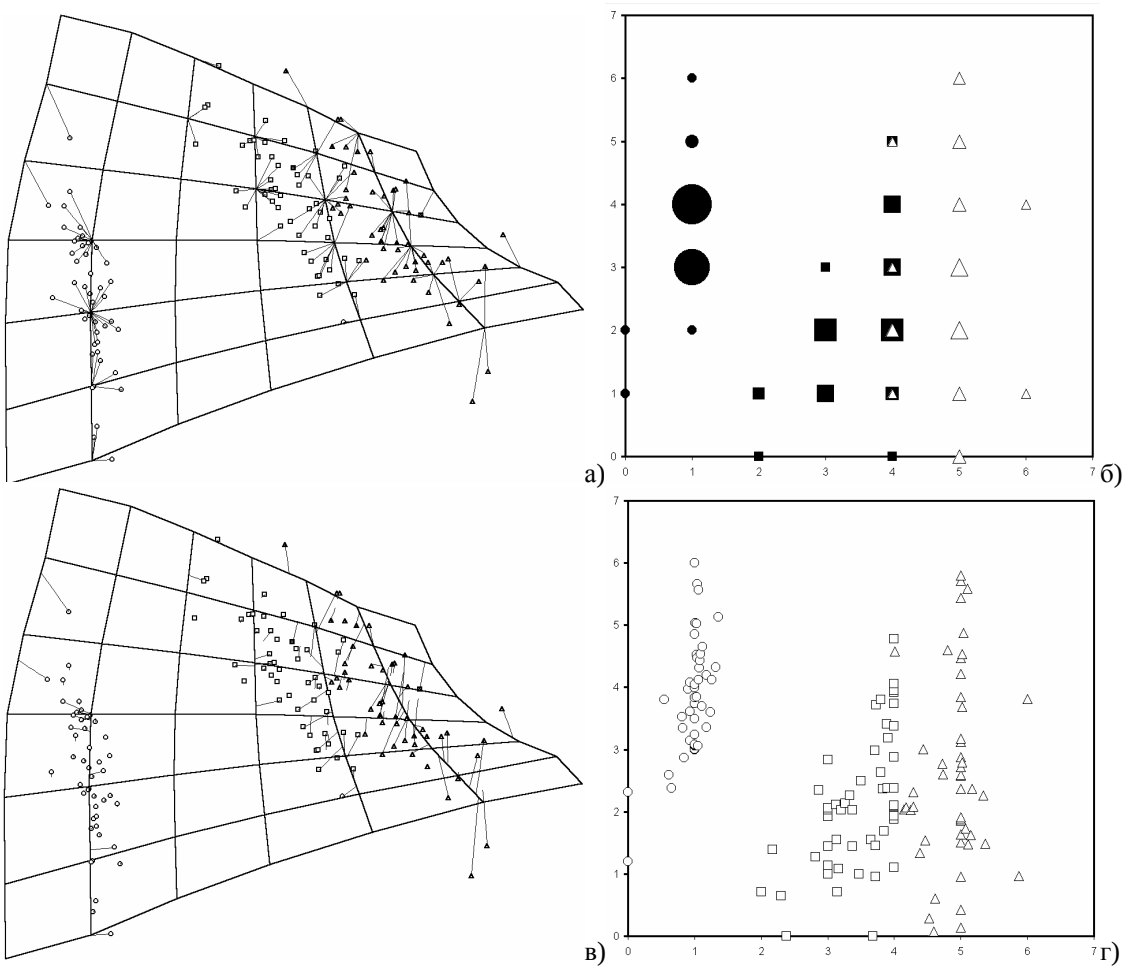


Рис. 16. Сравнение способов проецирования данных в ближайший узел и ближайшую точку карты (база данных по цветкам ириса).
а),б) проектирование в ближайший узел, на развертке условно изображено количество точек того или иного класса, попавших в узел; видно, что в четырех узлах после проектирования оказываются точки разных классов;
в),г) проектирование в ближайшую точку, на развертке видно, что разделение на классы можно сделать более четким (а также выделить «истинных» ренегатов класса).

Несмотря на то, что самоорганизующиеся карты Кохонена (СОК или SOM – Self-Organizing Maps или SOFM – Self-Organizing Feature Maps) изначально были описаны на нейросетевом языке, нам будет удобно рассматривать такие карты как двумерные сетки узлов, размещенных в многомерном пространстве, не прибегая к нейросетевой терминологии. Тем не менее, следует держать в уме то, что, если когда-нибудь алгоритм SOM будет воплощаться на аппаратном уровне, то для реализации высокоэффективных параллельных схем вычислений нужно будет вспомнить о изначальной нейросетевой архитектуре.

Итак, изначально SOM представляет из себя сетку из узлов, соединенный между собой связями. Кохонен рассматривал два варианта соединения узлов – в прямоугольную и гексагональную сетку (см. рис. 17) – отличие состоит в том, что в прямоугольной сетке каждый узел соединен с 4-мя соседними, а в гексагональной – с 6-ю ближайщими узлами. Таким образом, для двух таких сеток процесс построения SOM отличается лишь в том месте, где перебираются ближайшие к данному узлу соседи.

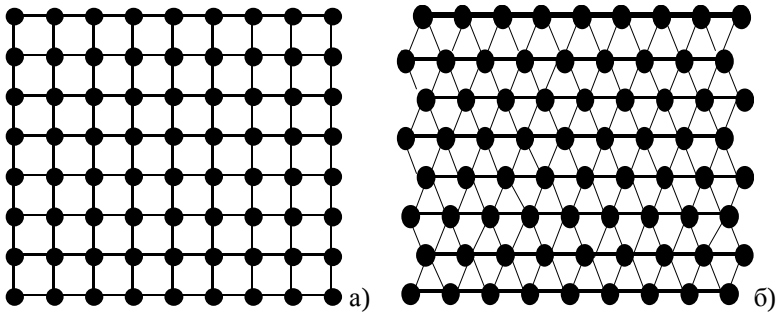


Рис. 17. Два варианта расположения узлов сетки SOM.
а) прямоугольная сетка – каждый узел (кроме краевых) имеет 4 ближайших соседа
б) гексагональная сетка – каждый узел (кроме краевых) имеет 6 ближайших соседей

Начальное положение сетки выбирается произвольным образом. А авторском пакете SOM_PAK предлагаются варианты случайного начального расположения узлов в пространстве и вариант расположения узлов в плоскости.

После этого узлы начинают перемещаться в пространстве согласно следующему алгоритму:

1. Случайным образом выбирается точка данных x.

2. Определяется ближайший к x узел карты r_{ij} (BMU – Best Matching Unit).

3. Этот узел перемещается на заданный шаг по направлению к x. Однако, он перемещается не один, а увлекает за собой определенное количе-

ство ближайших узлов из некоторой окрестности на карте. Поясним сказанное: если радиус окрестности равен 1, то вместе с ближайшим узлом r_{ij} по направлению к x двигаются 4 его соседа по карте в случае прямоугольной сетки и 6 соседей в случае гексагональной сетки.

В настройке карты различают два этапа – этап грубой (*ordering*) и этап тонкой (*fine-tuning*) настройки. На первом этапе выбираются большие значения окрестностей и движение узлов носит коллективный характер – в результате карта «расправляется» и грубым образом отражает структуру данных; на этапе тонкой настройки радиус окрестности равен 1–2 и настраиваются уже индивидуальные положения узлов.

О характере движения следует заметить следующее: обычно он настраивается так, чтобы во всем из всех двигающихся узлов наиболее сильно смещается центральный – ближайший к точке данных – узел, а остальные испытывают тем меньшие смещения, чем дальше они от центра узла (см. рис. 18)

Это соответствует так называемым затухающим функциям соседства (neighborhood function). Если это не так и все соседи из окружения испытывают равные смещения, то такая функция соседства называется пузырьковой (bubble) и для нее характерно большее число актов перемещения для обучения и менее гладкая сетка.

Кроме этого, величина смещения равномерно затухает со временем, то есть она велика в начале каждого из этапов обучения и близка к нулю в конце.

4. Алгоритм повторяется определенное число тактов. На первом этапе число тактов выбирается порядка тысяч, на втором – десятка тысяч (понятно, что число шагов может сильно изменяться в зависимости от задачи).

Отметим, что предложенный алгоритм не использует явно никакого критерия оптимизации. Хотя ясно, что, по крайней мере, будет уменьшаться среднее расстояние от каждой точки данных до ближайшего узла карты. Средний квадрат такого расстояния в пакете SOM_PAK служит критерием качества построенной карты. В этом пакете, как правило, строится несколько десятков карт, из которых выбирается лучшая согласно упомянутому критерию.

В результате действия алгоритма строится карта, то есть двумерная сетка узлов, размещенных в многомерном пространстве. Для того, чтобы изобразить их положение используются различные средства. Одно из них – такое раскрашивание карты, когда цвет отражает расстояние между узлами (см. рис.19) Узлы могут при этом маркироваться, если в исходном наборе данных имеются характерные точки данных с присвоенными метками – тогда эта метка ставится на узел, ближайший к данной точке.

Кроме того, сетка узлов может быть раскрашена согласно значению того или иного признака, причем это может быть признак, который не использовался при обучении в качестве координаты пространства (см. рис. 20)

Популярным способом изображения самих данных являются диаграммы Хинтона, когда на каждом узле сетки изображается квадрат, размер которого пропорционален числу точек данных, ближайших к данному узлу, а оттенок соответствует значению соответствующего отображаемого признака (см. рис. 21).

Если рассматривать самоорганизующуюся карту Кохонена как точечную аппроксимацию некоторого многообразия, то следует отметить следующее. Если считать, что узлы карты соответствуют равномерной сетке значений внутренних координат многообразия, то внутренняя метрика такого многообразия характеризуется сильной неравномерностью, – в местах скопления данных оказывается много узлов, и там многообразие получается «скомканным», а в местах, где данные отсутствуют узлов крайне мало, и там многообразие растянуто (см. рис. 26). Это означает, что после того, как многообразие будет «расправлено» на двумерной плоскости и изображено в равномерной сетке внутренних координат, точки данных, перенесенные на карту, окажутся расположены на ней более или менее равномерно. Второе замечание касается самого способа переноса точек из пространства на карту. Поскольку каждой точке данных сопоставляется ближайший узел карты, то, как уже упоминалось, такой вид проектирования является кусочно-постоянным. Это приводит к тому, что средний квадрат расстояния от точки данных до ее проекции на карте сильно зависит от количества узлов в сетке.

Примеры применения SOM.

1. Картографирование коллекций текстов.

Принципы представления большого количества текстов в виде частотных таблиц были рассмотрены в разделе 1.1. Визуализация таких таблиц с помощью SOM используется в Интернете для автоматического упорядочивания больших коллекций текстов.

Проект WebSOM визуализирует текстовую информацию из нескольких миллионов статей групп новостей UseNet. Полученная в результате карта показана на рис.22а и позволяет ориентироваться в море информации, собранной в этих статьях по тематикам. В отличие от простого каталога, такой способ представления рубрик обладает наглядностью и определенной ассоциативной непрерывностью – смежные темы занимают на карте смежные позиции.

Группой Терехова была предпринята аналогичная попытка визуализации тематического содержания статей и тезисов, опубликованных за не-

сколько лет на российских конференциях по нейроинформатике. В результате был создан удобный интерфейс (см. рис.22б), позволяющий пользователю по заданному ключу или автору получить визуальную картинку (диаграмму Хинтона) распределения интересующей его информации в базе статей и быстро сориентироваться в большом количестве публикаций.

- 2. Раскраска географической карты по экономической «схожести»

В Интернете (http://www.cis.hut.fi/nnrc/worldmap.html) есть пример использования SOM для построения некоего комплексного экономического показателя, по которому раскрашена обычная географическая карта. В результате станы со сходной экономической ситуацией изображены на карте цветами, близкими по спектру. (см. рис. 23)

- 3. Анализ основных экономических показателей крупнейших российских предприятий.

Шумским С.А., Кочкиным А.Н. проанализирована упомянутая в разделе 1.1 таблица основных экономических показателей для 200 крупнейших предприятий России. Для визуализации были использованы самоорганизующиеся карты Кохонена и диаграммы Хинтона.

- 4. Анализ фондового рынка

На рис.24 приведен пример анализа состояния фондового рынка с помощью самоорганизующихся карт, позволяющий одновременно оценить состояние рынка продаж по нескольким показателям (автором использовались Индекс релятивной силы (RSI), Степень изменчивости цены (Price ROC), Индикатор Вильямса (William's, %R), Индекс ценовых диапазонов (CCI), Осциллятор "Рэинбоу" (Rainbow Oscillator) и Стандартная ошибка (Standard Error)). Авторы примера (http://www.com2com.ru/dav/index.htm) считают, что такой подход является эффективным средством помощи при принятии решений на фондовом рынке («продавать/покупать»).

В литературе [51,57,59,61,63,69,77] приведено большое количество публикаций, в которых сообщается, что SOM применяется для автоматической классификации изображений, сжатия изображений, анализа временных рядов, задач ассоциативного поиска и т.д.

Со времени создания алгоритм SOM был подвергнут тщательному теоретическому исследованию. В литературе [60,62,64,71] описано большое количество модификаций первоначального алгоритма, краткий обзор

этих идей сделан в разделе 2.4. В основном, модификации приводят к тому, что задаваемая «руками» в первоначальном варианте алгоритма величина радиуса соседства определяется и настраивается автоматически с помощью тех или иных эвристических приемов, в результате чего самоорганизующиеся карты приобретают те или иные дополнительные свойства (регулярности, точности и т.д.).

1.5. Упругие карты

Многочисленные примеры использования идеологии SOM показывают, что визуализация данных при помощи вложенных в многомерное пространство двумерных сеток достаточно эффективна как средство анализа структуры многомерного облака данных.

Еще раз подчеркнем «технологические» особенности применения этой идеологии:

1) В настройке сети не используется оптимизация какого-либо функционала. Единственное, чего можно ожидать –это уменьшение среднего расстояния от точки данных до ближайшего к ней узла сетки. С одной стороны, тенденция к такому уменьшению заложена в самом алгоритме построения карты, с другой стороны утверждать достигает ли построенная сетка в действительности хотя бы локального минимума среднеквадратичной ошибки наверняка нельзя.

2) Проектирование данных осуществляется в ближайший узел карты, таким образом, этот узел является представителем своего локального мини-кластера данных.

3) Узлы построенной карты распределены в пространстве неравномерно. Если в облаке данных есть сгущения, то в них окажется больше узлов, чем в свободных от точек областях пространства. Таким образом, самоорганизующаяся карта осуществляет сокращение описания множества точек данных с помощью замены локальных сгущений в облаке данных на несколько «узлов-представителей», количество которых пропорционально размерам сгущения (*линейное векторное квантование* - LVQ) и одновременно связывает эти узлы в двумерную сетку, что позволяет расположить их на плоскости для визуального анализа отношений и расстояний между сгущениями.

Нами предлагается иная технология построения вложенных многообразий, которые мы будем называть *упругими картами*. Задачу построения вложенного многообразия, в отличие от SOM, мы поставим как оптимизационную, что соответствует общей методологической установке в прикладной статистике (см., например,[3,4]). То есть построенная карта будет решением задачи на оптимизацию заданного функционала от взаимного расположения карты и данных.

В принципе, в качестве критерия оптимальности мог бы быть использован любой из упомянутых проекционных индексов из методов целенаправленного проецирования (например, критерий стресса), однако, как мы уже указывали, во-первых, такая постановка задачи по существу нелинейна и ее применение требует использования градиентных процедур оптимизации, что само по себе составляет проблему для большого количества точек; во-вторых, как мы уже упоминали, карта как таковая строится уже после решения задачи оптимизации, и ее расположение в пространстве оказывается сильно неоднозначным, так как одни и те же результаты проецирования могут быть получены с использованием различных карт.

Попробуем сформулировать такой критерий оптимальности, в который бы входили не начальные и конечные положения точек (до и после проецирования на карту), а положения самих узлов карты в пространстве относительно данных. Тогда существенно снизится размерность задачи оптимизации – с размерности mN в случае задач целенаправленного проецирования до mpq (p, q – число узлов прямоугольной сетки по горизонтали и вертикали, m – размерность пространства, N – число точек). Кроме того, вид функционала можно попробовать сделать квадратичным по положению координат узлов карты для того, чтобы в результате решать систему линейных уравнений для нахождения минимума критерия.

Ясно, что в критерий должно входить среднее расстояние от точки данных до ближайшего узла карты. Мы должны минимизировать его для того, чтобы карта моделировала данные. С одной стороны, такой критерий в случае нормального распределения заставит все узлы карты разместиться в плоскости первых двух главных компонент, и, таким образом, построенная карта может служить обобщением метода главных компонент . С другой стороны, если в качестве меры длины выбрать обычное евклидово расстояние, то эта часть критерия оптимальности будет квадратична по координатам положения узлов, что весьма желательно.

Однако минимизировать указанный критерий можно бесконечным числом способов, строя в том числе и такие карты, узлы которых будут совершенно неупорядочены. В самоорганизующихся картах упорядочивание достигается за счет того, что между узлами существуют связи и каждый узел, двигаясь в пространстве, подтягивает за собой соседей. На основании этого соображения можно добавить в критерий требование упругости карты. Что это означает?

Вообразим себе упругую двумерную поверхность – например, кусок упругой пластинки. Такая пластина при различных деформациях стремиться восстановить свою первоначальную форму. Однако, деформировать пластину можно двумя способами – растягивая ее «вдоль» и изгибая «поперек» (см. рис.25) – и в одном случае она стремиться сохранить свою длину, в другом – свою плоскую форму. Назовем возникающие при деформациях силы в пластике упругостью по отношению к растяжению и упругостью по отношению к изгибу.

Теперь, если мы потребуем, чтобы наша сетка обладала обоими этими свойствами, то в минимизируемый критерий необходимо добавить меру суммарного растяжения сетки и меру суммарного изгиба. Такие меры в самом простом варианте описания упругих сил также оказываются квадратичными по отношению к координатам положения узлов сетки в пространстве.

Складывая вместе все три упомянутые меры (средний квадрат расстояния до узла и две меры упругости) с определенными весами, мы получаем общий критерий, благодаря которому сетка, с одной стороны, будет притягиваться к точкам данных, с другой – стремиться минимизировать свое растяжение и принять максимально гладкую форму (стать более регулярной). Конкретный вид рассмотренного критерия и алгоритм его минимизации подробно рассмотрены в разделе 2.5.

У построенной сетки остаются неопределенными два параметра – веса суммы – они могут быть интерпретированы как коэффициенты упругости сетки по отношению к растяжению и изгибу. Их приходится задавать «руками», но можно построить и такую процедуру настройки этих параметров, чтобы сетка приняла желаемый вид. С одной стороны, чем более упруга сетка, тем более гладкую модель данных она собой представляет, но и тем хуже она описывает малые отклонения от предполагаемого закона; с другой – чем менее упруга сетка, тем она точнее описывает данные, но при этом воспроизводятся и все случайные шумы, которые обычно присущи реальным данным, при этом ухудшается способность модели к обобщению информации.

Самой простой идеей настройки коэффициентов упругости является последовательное их уменьшение от больших значений к малым до тех пор, пока не будет достигнута необходимая точность. При больших значениях коэффициентов упругости узлы карты практически находятся в одной плоскости, и это будет плоскость главных компонент, на которой, кстати, их можно изначально и разместить. Далее карта будет приобретать криволинейную форму, аппроксимируя распределение данных.

Что будет, если упругость карты станет равна нулю? Рассмотрим те точки, которые окружают определенный узел карты и находятся ближе к нему, чем к любым другим узлам. Как и прежде, назовем множество таких точек *таксоном* данного узла. Если таксон не содержит ни одной точки, то узел останется на месте, если таксон состоит из единственной точки, то узел переместится в нее, если таксон состоит из нескольких точек, то узел разместится в точке среднего значения всех координат точек таксона (см. рис.25б) В этом смысле карта с нулевой упругостью схожа с картой Кохонена – положения ее узлов совпадают с центрами локальных сгущений. Однако, упругая карта в силу способа построения отличается от самоорганизующейся карты – так число узлов, размещенных в областях сгущения данных не будет пропорционально мощности сгущений. Дело в том, что некоторым узлам всегда будет «энергетически выгодно» расположиться в

пространстве между сгущениями – и, таким образом, сетка окажется более-менее равномерной, а не так сильно деформированной, как в случае SOM.

Так как узлы сетки притягиваются не только к точкам данных, но и к друг другу, то некоторые данные могут располагаться в пространстве между узлами, на сравнительно большом удалении от них. Отсюда следует, что процедура проецирования данных в ближайший узел сетки может давать большие значения ошибки, по сравнению с Кохоненовскими картами. Таким образом, для упругих карт особенно актуально становится использование кусочно-линейных методов проектирования, – например, проектирование в ближайшую точку (не узел!!) карты. Но, как уже было сказано, для этого необходимо интерполировать многообразие в промежутках между узлами, например, это можно сделать кусочно-линейным способом, сделав «граненую» карту (см. рис.26), для которой найти ближайшую точку достаточно просто.

1.6. Картографирование данных

После того, как карта построена, и точки данных перенесены из пространства признаков на поверхность карты, можно пользоваться ее изначальной двумерностью, расправив ее складки и развернув на плоскости. Теперь каждая точка данных имеет две координаты во внутренней системе координат на карте.

Плоскую карту можно изобразить двумя способами (см. рис.26). В первом можно попытаться максимально воспроизвести те расстояния между узлами, которые были в исходном пространстве, получив при этом двумерную *криволинейную координатную сетку*. Понятно, что сделать это совсем без искажений не удастся, поскольку исходная сетка вложена в многомерное пространство (расправить сетку без искажений нельзя именно на плоскости, известно, что в евклидово трехмерное пространство можно вложить двумерную поверхность с произвольной метрикой). Во втором можно изображать проекции данных в исходных внутренних координатах. Тогда «развернутая» карта просто имеет вид прямоугольника.

Таким образом, мы получим своеобразную подложку, которая сформируется при помощи данных под влиянием двух конкурирующих тенденций – стремлении узлов притянуться к данным, и стремлении к минимальному растяжению и изгибу сетки. На этой подложке мы можем изображать различную информацию. С помощью линий уровня и различных раскрасок можно изображать значения тех или иных интересующих исследователя величин и объектов. Каждый из способов раскраски предстанет в конечном виде как *информационный слой*, являющийся аналогом слоя в ставших традиционными ГИС-технологиях (ГИС – *геоинформационные системы* – средства компьютерного представления информации, привязанной к географической карте). Разница состоит лишь в том, что вместо обычной гео-

графической карты используются подложка – двумерное многообразие, вложенное в многомерное пространство данных. Подобно тому, как на географической карте рядом оказываются объекты с близкими значениями географических координат, так и на построенной подложке рядом располагаются объекты с близкими значениями признаков в исходном пространстве. Это позволяет «один к одному» применять весь богатейший арсенал средств ГИС. В результате для применения ГИС-технологий открывается *возможность картографирования данных произвольной природы*, представленной в виде таблиц. Окончательный результат применения ГИС-идеологии – атлас тематических раскрасок, дающих представление о внутренней структуре данных.

Какие раскраски – информационные слои – могут быть включены в этот атлас? Или, другими словами, что можно изобразить на построенной карте?

Во-первых, можно изобразить *сами данные*. Данные можно изображать точками, однако, эффективно иметь возможность отображать в местах проекций точек данных разнообразную связанную с ними информацию. Для «увеличения размерности» точек данных могут быть использованы следующие приемы:

а) использование цвета, размера и формы для изображения точек данных; это дает возможность отражать три дополнительных измерения, связанных с точками: цвет и размер позволяют изображать количественные (непрерывные) шкалы, форма – номинальные шкалы признаков;

б) использование сложных изображений – например, точку можно изображать круговой диаграммой, на которой цветами изображено соотношение между значениями координат признаков, а размер отражает абсолютные величины координат (см. рис. 27)

Во-вторых, на карте с помощью линий уровня можно изображать значения любого функционала, заданного в пространстве признаков. Естественно, эти значения будут вычисляться в точках размещения самой карты. В качестве таких функционалов можно использовать следующие величины:

а) значения координаты какого-либо признака – это самый простой тип раскраски, который позволяет выделить на карте области с определенным значением того или иного признака и размещение в них точек данных; понятно, что таких раскрасок будет столько, сколько признаков было в исходном пространстве, то есть число раскрасок по признакам равно размерности пространства данных, и для случая высокой размерности исследователь может просто запутаться в большом количестве картинок. Для того, чтобы предоставить исследователю наиболее содержательные и информативные раскраски, необходимо иметь возможность предварительного отбора признаков по их значимости;

б) простые функции от пары признаков – например, их разность или отношение; смысл таких раскрасок может заключаться в том, чтобы сравнить раскраски по значениям нескольких признаков; например, если раскраски по значениям координат двух признаков схожи, то это указывает на их сильную скореллированность; и наоборот, сравнение раскрасок позволяет выделить те области пространства и точки данных, для которых корреляционная зависимость нарушается;

в) сложные функционалы от координат признаков – например, в каждой точке карты каким-либо непараметрическим способом можно оценить многомерную плотность распределения данных. При этом можно изображать раскраску, отражающую распределение как всего облака данных, так и какого-либо его подмножества, или более содержательную раскраску по относительной плотности подмножества, равной отношению значения плотности подмножества к общей плотности всего множества;

> ✂ Смысл вычисления относительной плотности подмножества заключается в следующем. Если в какой-либо области пространства содержится мало точек какого-то класса, это еще не означает, что для этой области вообще вероятность появления точек этого класса мала, – возможно, что в обучающем множестве также находилось мало данных из этой области. Говоря о «малой» или «большой» плотности подмножества в какой-либо точке, всегда необходимо указывать, с чем эта плотность сравнивается. ✂

С другой стороны, плотность данных и их подмножеств можно рассчитывать уже в двумерном пространстве самой карты – такую плотность можно назвать двумерной.

г) на карте в виде непрерывных раскрасок можно изображать свойства самой карты; например, цветом можно изобразить области ее наибольшего растяжения и сжатия в исходном пространстве; в частности, можно изобразить значения метрических коэффициентов на поверхности карты (понятно, что в случае кусочно-линейной карты эти значения постоянны в пределах одной плоской грани карты);

д) на карте можно изображать разнообразные поля раскрасок, иллюстрирующие взаимные свойства карты и данных, отражающие способность построенной карты служить моделью данных; одна из таких раскрасок – расстояние от точки карты в исходном пространстве признаков до ближайшей точки данных; так на карте можно увидеть 1) насколько сильно узлы карты удалены друг от друга и от скоплений данных и 2) те области пространства, где карта неплотно прилегает к данным и, следовательно, не может служить в этих областях хорошей моделью множества точек.

Другой вариант раскраски поможет исследователю оценить, насколько хорошо сохраняются отношения соседства после проецирования точек данных на карту. Дело в том, что любое проектирование в простран-

ство меньшей размерности чревато *изменением топологических особенностей* исходного множества – далекие точки в многомерном пространстве могут оказаться близкими на двумерной карте, и, наоборот, точки, близкие в исходном пространстве могут оказаться разнесены на карте на далекие расстояния (см. рис. 28). Такие несоответствия можно назвать *искажениями структуры данных*.

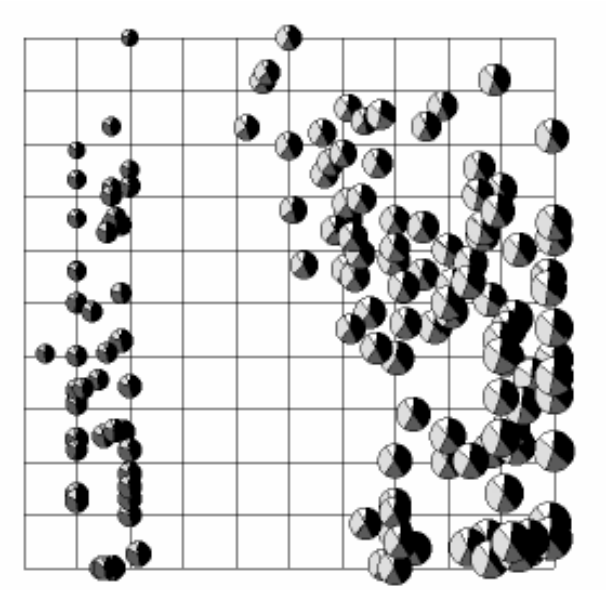


Рис. 27. Использование вида точек-проекций данных для увеличения «размерности» точек данных.

На рисунке изображены проекции точек данных для визуализации базы данных «Ирис». Каждая точка данных представлена в виде круговой диаграммы. Черный цвет изображает значение признака «длина лепестка», темно-серый – «ширина лепестка», светло-серый – «длина чашелистика», белый – «ширина чашелистика». Площадь соответствующего сектора пропорциональна значению признака. Из вида проекций наглядно видно, например, что класс Iris-Setosa – облако данных слева на карте – характеризуются маленькими цветками с относительно узкими чашелистиками и длинными лепестками.

Один из способов наглядно изобразить, каким образом искажается структура данных после проецирования на двумерную поверхность заключается в следующем. В каждой точке карты вычисляется множество K1 ближайших к данной точке соседних точек в исходном пространстве и множество K2 ближайших соседей в двумерном пространстве карты после проецирования. Мощность пересечения K1 и K2 (число совпадающих точек) может служить мерой сохранения отношений соседства между точками данных после их проецирования. Чем меньше это число, тем хуже передаются топологические особенности исследуемого множества объектов. С помощью таких раскрасок исследователь может выделить те области пространства, в которых карта в упомянутом смысле плохо отражает структуру данных. Это означает, что в таких областях моделирование и ви-

зуализация данных с помощью двумерных поверхностей не имеет большого смысла.

✂ Некоторыми авторами специально исследовалась способность карт Кохонена к отражению топологической структуры исходного множества данных. Например, в [64] предлагается для оценивания качества построения самоорганизующейся карты использовать наряду с MSE (Mean Square Error – среднеквадратичная ошибка) еще и так называемую *топографическую ошибку*. Вычисляется она следующим образом: Для каждой точки данных находится ближайший узел на карте r_1 и второй по близости – узел r_2. Если на карте эти два узла являются смежными, то такая точка пропускается, если нет – то считается, что проекция такой точки *неустойчива* – при небольшом изменении карты или положения точки перемещение ее проекции на карте может произойти скачком. Относительное число таких неустойчивых точек и называется топографической ошибкой. Для каждой из неустойчивых точек можно оценить «меру неустойчивости» – выяснив, насколько далеко отстоят друг от друга узлы r_1 и r_2. Построив гистограмму распределения неустойчивых точек по «мере неустойчивости», получаем изображение так называемой *функции топографической ошибки*. ✂

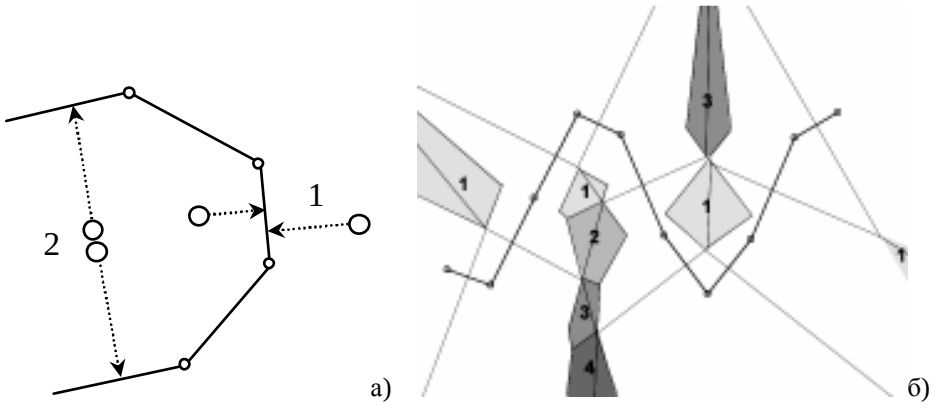


Рис. 28. Искажения структуры данных при проецировании на карту.
а) 1 - далекие точки в пространстве могут оказаться на карте рядом (плохое разрешение); 2 – близкие точки могут оказаться разнесенными на карте на большое расстояние (искажение топологии данных);
б) искажение топологии при проецировании: толстая линия – одномерная карта на плоскости, если точка данных находится в закрашенной области, то она окажется неустойчивой – ближайший (BMU) и второй по близости (BMU2) узел карты не являются смежными, цифрой обозначена степень неустойчивости – насколько далеки оказались друг от друга узлы BMU и BMU2; на картину областей неустойчивости тонкими линиями нанесены границы ячеек Вороного.

1.7. Мультикартирование и восстановление данных

Мы уже подчеркивали, что при создании модели данных необходимо найти компромисс между двумя свойствами модели – свойством воспроизводить исходные данные с высокой точностью и свойством иметь обобщающую способность, то есть отражать не случайные, а существенные характеристики набора данных, что позволяет использовать модель для тех данных, которые не участвовали в настройке модели. На языке упругих карт это означает, что, с одной стороны, карта должна быть достаточно «мягкой», чтобы близко прилегать к точкам данных и аппроксимировать их с достаточной точностью, с другой, карта должна быть упругой, чтобы быть гладкой и не аппроксимировать случайный шум (но при этом снижается точность аппроксимации). Если карта используется не только для визуализации данных, но и для построения регрессионных зависимостей одних признаков от других, или для прогнозирования (предсказания значений признаков), или для восстановления пробелов в данных, о чем речь пойдет ниже, то весьма перспективной может оказаться применение такого приема: по данным составляются *несколько* карт, каждая из которых имеет достаточно гладкую поверхность, но первая из них картографирует и аппроксимирует сами данные, вторая – отклонения от первой модели, то есть множество векторов, которые начинаются в точках данных и заканчиваются в точках соответствующих проекций данных на первую карту. Эти отклонения имеют две составляющие – случайный шум и неточности первой модели. Для построения третьей карты используются отклонения от проекций первых отклонений и так далее.

Рассмотрим чуть подробнее, как будет выглядеть карта первых отклонений. Если большинство точек все-таки более-менее адекватно описываются первой картой, то большая часть данных будет тяготеть к нулю новых координат (в пространстве отклонений). Если все отклонения могут быть отнесены на счет случайного шума, то закон распределения отклонений будет близок к нормальному (это, например, часто предполагается в традиционном факторном анализе).

В результате применения процедуры *мультикартирования* пользователь получает набор карт – *видов* данных, которые можно обозначить следующим образом: «вид данных», «вид первых остатков», «вид вторых остатков» и так далее. На рис.29 показан пример такого мультикартирования.

Какой смысл имеет построение нескольких карт? Во-первых, пользователь имеет возможность выделить на карте «аномальные» объекты, которые плохо описываются с помощью модели и проанализировать из расположение в пространстве остатков с помощью информационных раскрасок. Во-вторых, использование нескольких карт позволяет существенно увеличить точность описания данных, не делая при этом карту данных слишком «неровной». Таким образом, мы, с одной стороны, получаем хорошую точность описания данных, с другой – хорошую обобщающую спо-

собность построенной модели, поскольку моделирующие многообразия могут быть сделаны достаточно гладкими.

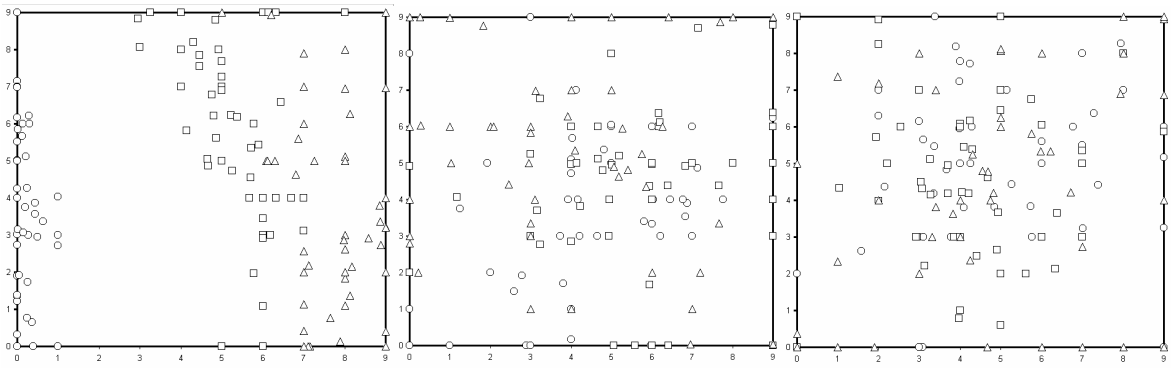


Рис. 29. Пример мультикартирования данных (база Iris)

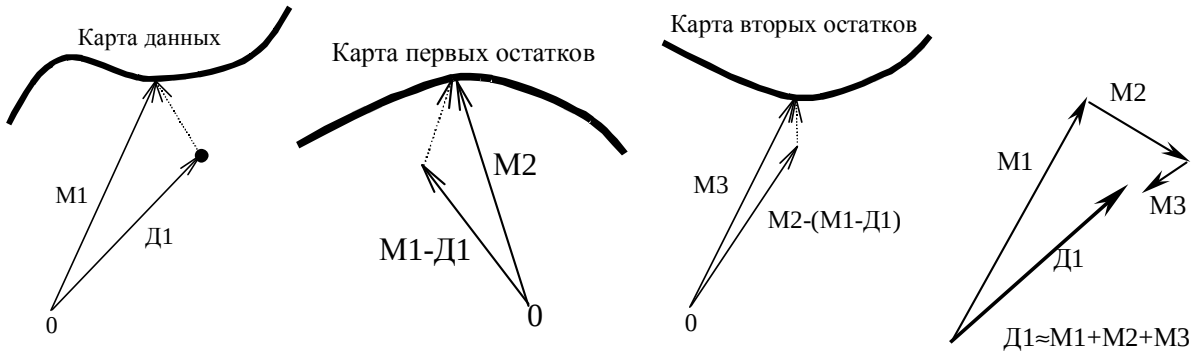


Рис. 30. Восстановление данных с помощью нескольких карт

Иными словами, мы по отдельности создаем гладкую модель самих данных, затем гладкую модель помех и так далее. При этом, в отличие от стандартного факторного анализа, не делается никаких априорных представлений о природе и структуре отклонений от модели – они описываются «как есть».

Повышение обобщающей способности модели открывает возможность правдоподобного восстановления проекций данных. На рис.30 показано каким образом вектор данных Д1, содержащий пропущенные значения признаков может быть восстановлен последовательностью моделирующих векторов М1, М2, М3 …, которая строится по последовательности карт. Здесь существенную роль играет то обстоятельство, что мы можем проецировать вектор Д1 на карту, несмотря на то, что он содержит пропущенные значения признаков.

1.8. Особенности и ограничения подхода

1.8.1. Экстраполяция и интерполяция карты

Построенная карта, в отличие от плоскости первых главных компонент, представляет собой *ограниченное* многообразие. Его ограниченность связана с самим способом построения (нам было бы трудно оперировать с бесконечной сеткой). Поскольку карта стремится расположиться как можно плотнее к данным, в результате она представляет собой кусок криволинейной поверхности, расположенный *внутри* облака точек данных.

Из-за этой особенности карты неизбежно возникают краевые эффекты. На рис.31 видно, что после проецирования много точек оказываются расположенными на границе карты из-за того, что ближайшими точками карты для точек, лежащих на периферии облака данных, будут граничные узлы и ребра. Это существенно искажает вид таких периферийных структур после проецирования на карту – создается ложное впечатление, что данные группируются в периферийных областях.

В связи с этим замечанием к технологии построения карты желательно было бы добавить различные способы экстраполяции карты на окрестность таким образом, чтобы свести к минимуму нежелательные краевые эффекты (карта должна частично выходить за пределы группирования данных).

Самый простой способ экстраполяции – линейный, когда карта продолжается за свои пределы вдоль направлений, которые задаются ребрами, прилегающими к краям карты.

Более интересными могут оказаться нелинейные способы экстраполяции. Например, для экстраполяции можно использовать двумерную формулу Карлемана. Однако, для нее существенным является вопрос о граничных условиях – о форме поведения экстраполируемой поверхности вдали от карты. Два простых варианта – в первом в качестве асимптотического условия используется плоскость первых главных компонент, во втором – довольно своеобразный способ соединения границ карты в одной точке – например, в какой-либо удаленной точке или в точке среднего значения координат точек данных. В последнем случае карта «замыкается», то есть все граничные точки карты мысленно склеиваются, что в некоторых ситуациях может иметь физический смысл.

На рис.31 показаны разные способы экстраполяции карты на окрестность и получаемые в результате проекции точек данных.

Необходимость в интерполяции карты (добавлении дополнительных узлов в аппроксимирующую сетку) возникает в случае, когда карта представляет из себя кусочно-линейное многообразие. Дело в том, что для сильно «угловатой» карты размеры областей пространства, для ко-

торых ближайшей точкой карты является узел, велики и многие точки проектируются в результате в этот узел, в результате чего ухудшается разрешающая способность карты – точки из упомянутых областей сливаются в одну проекцию. Разрешающую способность можно повысить, сделав карту более гладкой, введя в промежутки дополнительные узлы.

Также можно выделить два способа интерполяции – линейный, в результате которого треугольники, образующие грани карты разбиваются на несколько маленьких треугольников и нелинейные, в результате применения которых новые узлы располагаются на нелинейной поверхности, полученной в результате применения различных интерполяционных формул. И в этом случае хорошие результаты показывает применение двумерной формулы Карлемана. Интерполяция с использованием формул Карлемана, помимо прочего, является в некотором роде оптимальной.

На рис.31 показаны варианты сглаживания карты и результирующие виды спроецированных данных.

1.8.2. Качество визуализации и сложные распределения данных

Понятно, что представление данных с помощью вложенных двумерных карт тем адекватнее отражает реальные структуры, содержащиеся в данных, чем ближе эффективная размерность облака данных к двум. В этом смысле хуже всего картографируются данные, распределение которых близко к равномерному.

Не так просто дать количественные оценки качества картографирования данных. Одна из оценок очевидна – это среднее расстояние от точки данных до ближайшего к ней узла (MSE). Уточненным вариантом этой оценки является среднее расстояние от точки данных до ее проекции (назовем ее MSPE), что более естественно в случае некусочно-постоянных способов проецирования.

Вторая оценка характеризует устойчивость проецирования по отношению к малым изменениям положения точки данных в пространстве признаков. Выше мы уже вводили величину, характеризующую эту устойчивость – это топографическая ошибка (TE).

Третью оценку можно получить, выясняя различия в двух списках – M ближайших соседей для каждой точки в пространстве признаков и M ближайших соседей в двумерном пространстве карты. Функция зависимости среднего числа различий в двух таких списках от M дает представление о сохранении отношений соседства между точками после проецирования на карту. Обозначим эту оценку через NRE (Neighborhood Relation Error).

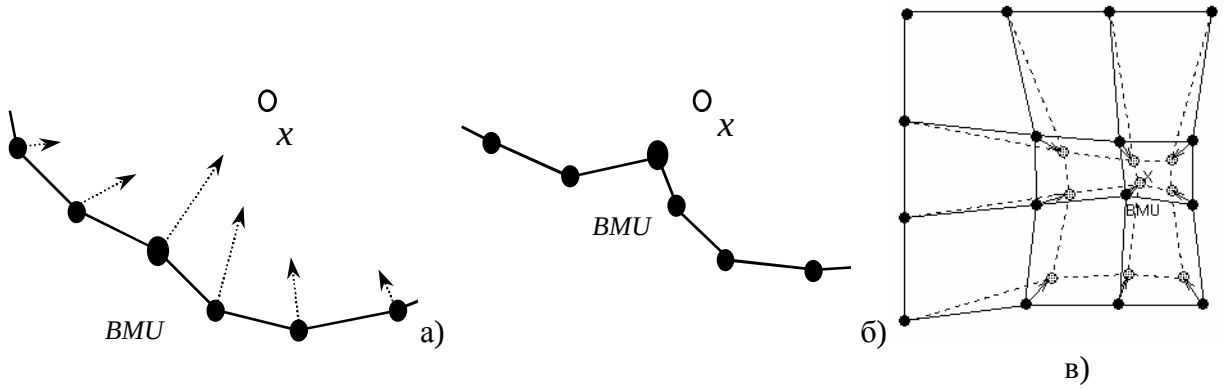


Рис. 18. Иллюстрация работы алгоритма SOM
а),б) результат действия одной итерации в случае одномерной сетки узлов;
в) результат действия одной итерации в случае двумерной сетки узлов.

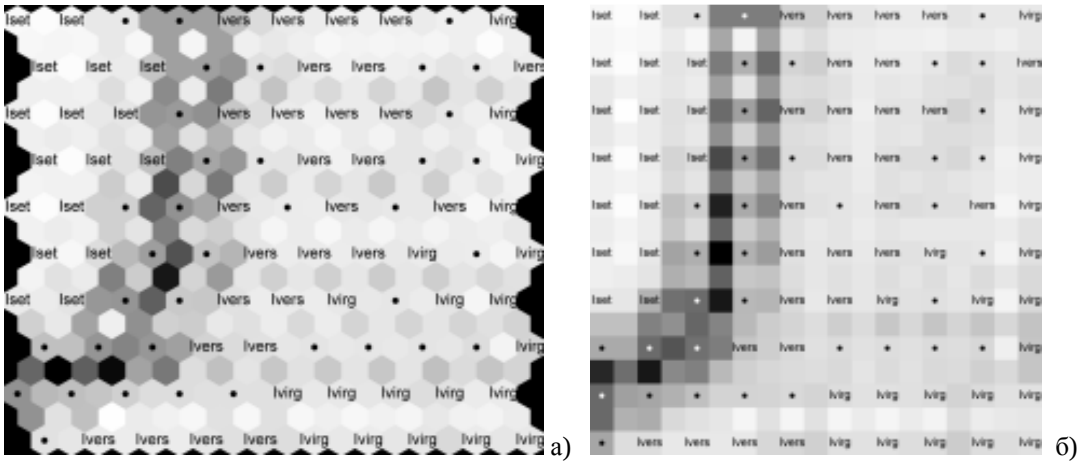
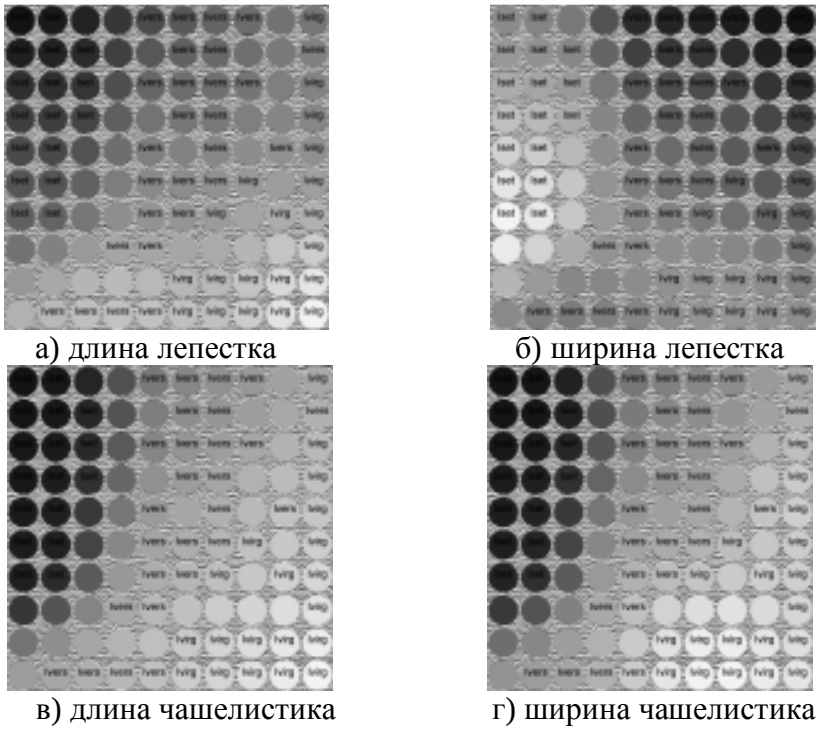


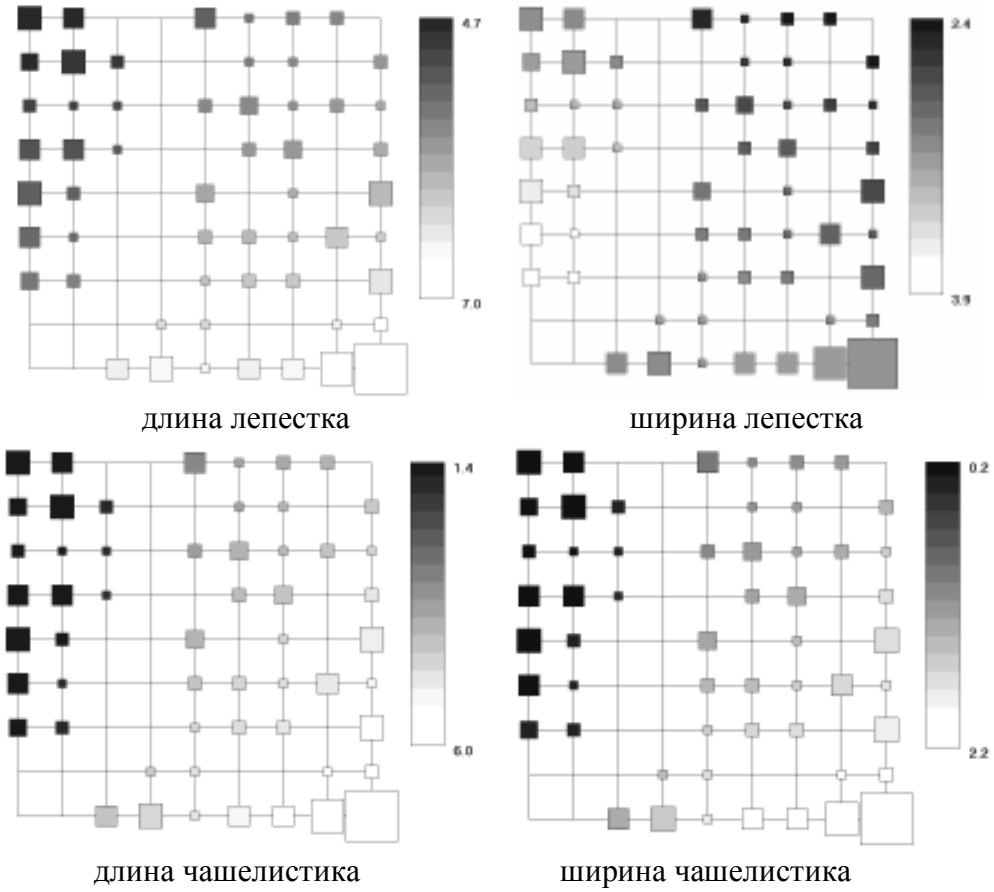
Рис. 19. Визуализация данных в виде U-матрицы.
Точками и метками обозначены узлы, метки показывают точки какого класса оказались в узле. Оттенок ячейки, расположенной между двумя узлами, отражает расстояние между узлами в исходном пространстве. Более темный оттенок соответствует большему расстоянию. Оттенок самого узла вычисляется с помощью усреднения. Из раскраски можно сделать вывод о том, что класс Iris-setosa хорошо пространственно отделен от двух других классов.
а) случай гексагональной сетки; б) случай прямоугольной сетки.



а) длина лепестка б) ширина лепестка

в) длина чашелистика г) ширина чашелистика

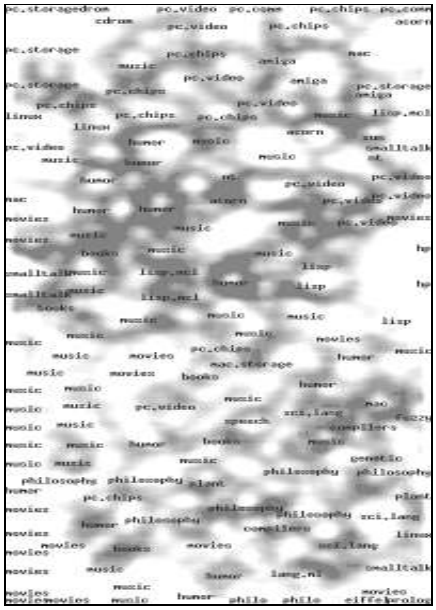
Рис. 20. Представление значений признаков с помощью раскрасок. Более темный оттенок соответствует меньшим значениям признака.



длина лепестка ширина лепестка

длина чашелистика ширина чашелистика

Рис. 21. Визуализация данных с помощью диаграмм Хинтона. Чем больше точек попадает в узел, тем больше соответствующий размер квадрата. Оттенок квадрата соответствует значению соответствующего признака



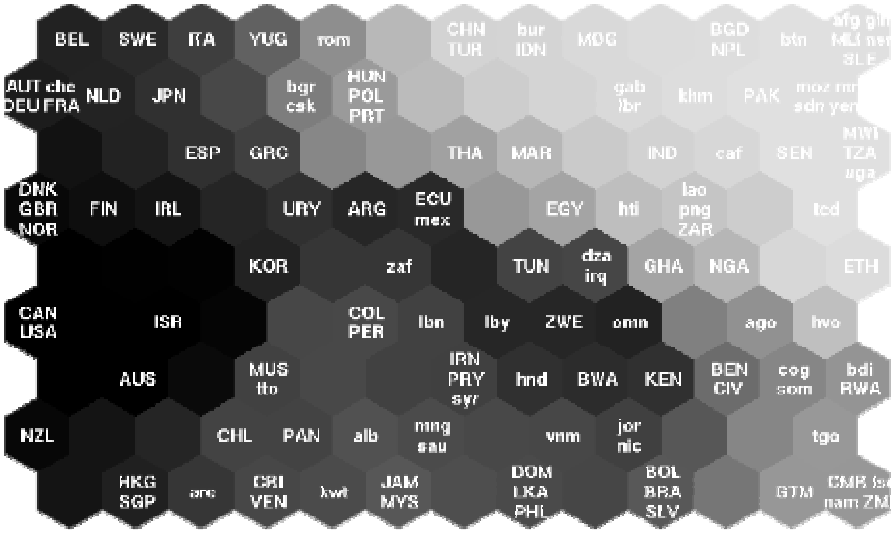
а)

б)

Рис. 22. Примеры применения SOM для картографирования больших текстовых коллекций.

а) Проект WebSOM – картографирование содержания более полутора миллионов статей UseNet. На рисунке показана карта «верхнего уровня» – своеобразный двумерный каталог статей, в котором статьи, близкие по тематике расположены по возможности в близких областях. Метки обозначают наиболее часто употребимое слово в данной области тематической карты. Цвет отражает насыщенность той или иной темы статьями. Щелкнув по соответствующей области, можно получить карту более низкого уровня, где информация представлена более подробно для выбранной темы. Таким образом, можно добраться до карты самого низкого уровня, где можно уже будет просматривать содержание отдельных статей. Такое построение карт называется «иерархическим».

б) Проект ConfWeb – визуализация в виде диаграммы Хинтона содержания нескольких сотен статей и тезисов докладов, представленных на российских конференциях по нейроинформатике за несколько лет. Изначально статьи представлены в виде одноцветной диаграммы Хинтона. В режиме поиска «По автору» подсвечиваются те квадраты, которые отображают те кластеры данных, в которых фигурирует имя автора, давая своеобразную «картину интересов» автора. В режиме поиска «По ассоциациям» подсвечивается тот квадрат, в кластере которого содержится введенное слово. Щелкнув по любому квадрату, в правом фрейме можно получить заголовки и авторов статей, вошедших в кластер.



а)

б)

Рис. 23. Раскраска географической карты с помощью SOM.

а) Карта Кохонена, построенная по нескольким десяткам экономических и социальных показателей разных стран (по странам СНГ у создателей карты данных не было). После построения гексагональная сетка не была окрашена, на нее были лишь нанесены метки стран, причем страны, оказавшиеся рядом на карте обладают сходными показателями. Затем на карту Кохонена был наложен двумерный цветной спектр (на рисунке нет возможности показать цветную раскраску), в результате чего каждый узел получил свой цвет, причем соседние узлы получили близкие по спектру цвета.

б) Цвета узлов были использованы для раскраски мировой карты по «похожести» показателей.

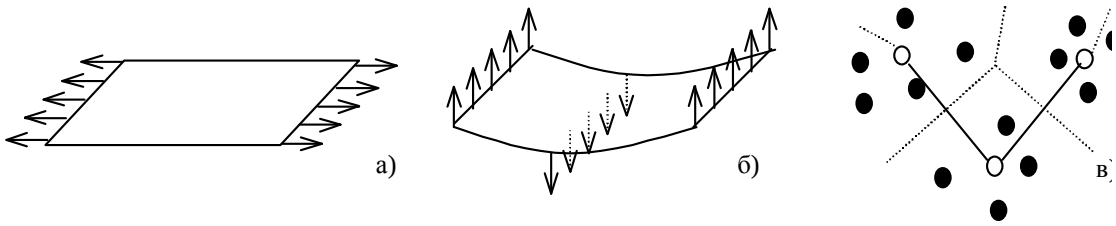


Рис. 25.
а) силы, растягивающие упругую пластину; б) силы, изгибающие упругую пластину; в) одномерный случай абсолютно мягкой карты – каждый узел карты располагается в центре «таксона», пунктиром изображены границы зон, в пределах которых точки являются ближайшими к данному узлу – так называемых *ячеек Вороного*.

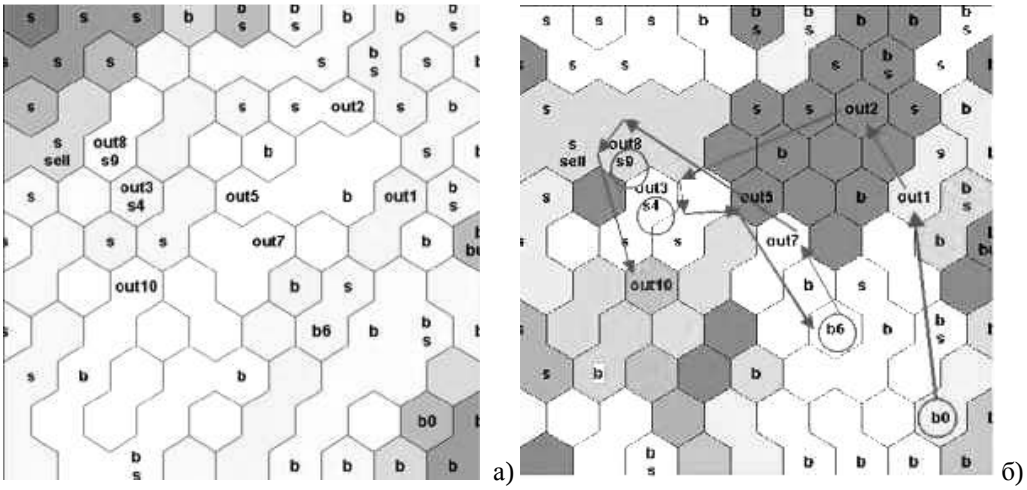


Рис. 24. Применение технологии SOM для анализа состояния фондового рынка.
а) Карта Кохонена, построенная по ежесуточным значениям основных показателей фондового рынка за некоторый промежуток времени. Светлым цветом изображены узлы, расположенные ближе к центру многомерного облака данных, темным – «периферийные» узлы. Значки «b», «s», «out» обозначают состояния рынка, в которых предпочтительнее покупка (buy), продажа (sell) и бездействие соответственно. В узлах, где одновременно находятся значки «b» и «s» возможна и покупка, и продажа. Таким образом, на карту нанесен некоторый «опыт» покупок и продаж.
б) Раскраска несколько иная. Светлым обозначены узлы, расположенные ближе к сгущениям данных. Стрелками изображены состояния рынка за 11 последовательных дней. Кружками отмечены дни, когда инвестором совершались сделки. Таким образом, инвестор может мгновенно сопоставлять свою деятельность с накопленным опытом рынка, принимая более оперативные решения.

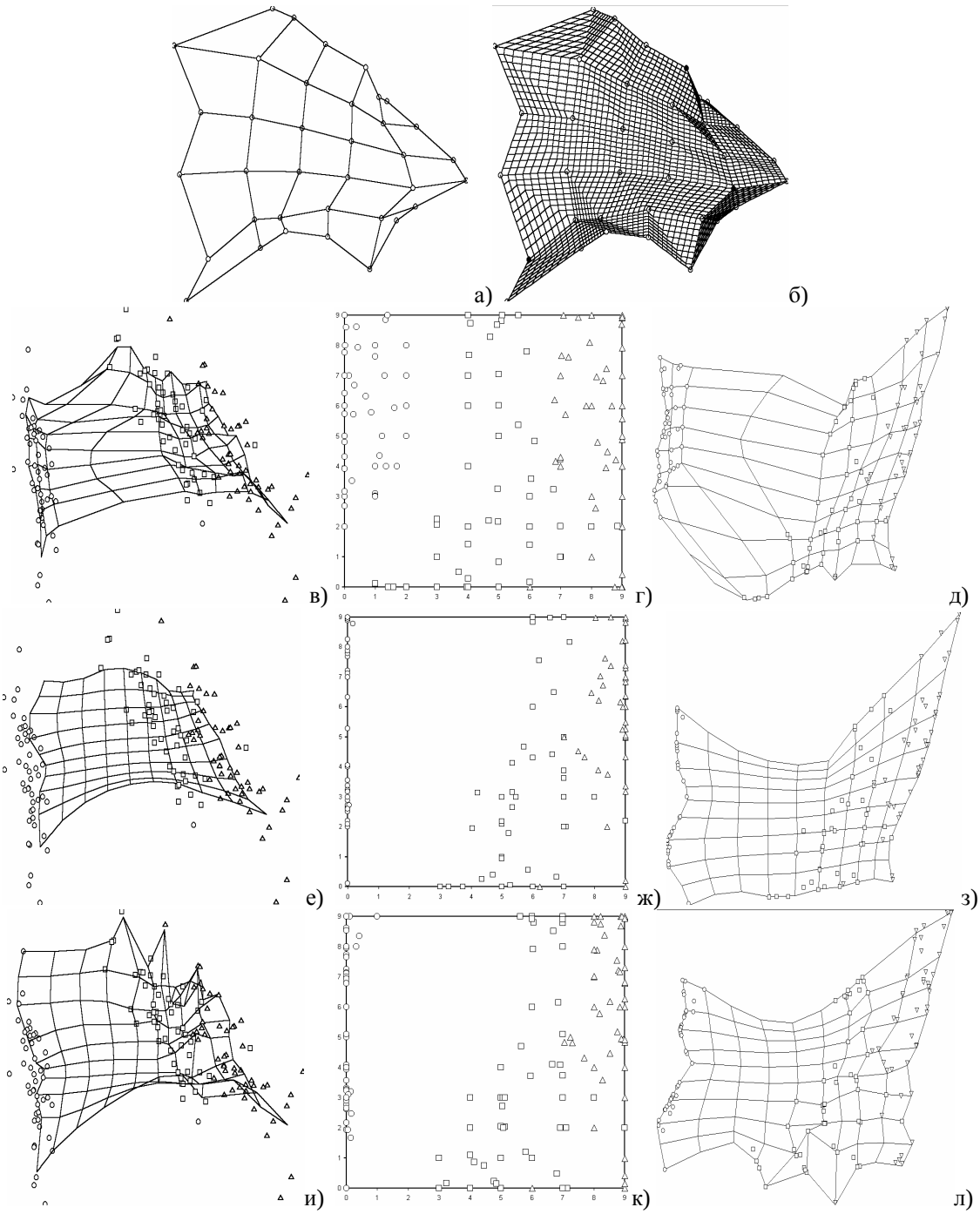


Рис. 26. Иллюстрации к работе метода упругих карт.
а),б) кусочно-линейный способ доопределения карты до многообразия («граненая» карта);
в),г) карта, построенная в результате работы алгоритма SOM и результирующие проекции данных; видно, что сетка сильно растянута в середине, где данных нет, в результате проекции располагаются на карте более равномерно, чем в исходном пространстве (кластерная структура «смазывается»);
е),ж) упругая карта с относительно большими значениями коэффициентов упругости, видно, что сетка более равномерная, в результате проекции расположены более адекватно; на проекциях виден недостаток упругой карты – много проекций оказались на краю карты (об этом подробнее в разделе 1.8.1);
и),к) упругая карта с относительно малыми значениями коэффициентов упругости;
д),з),л) соответствующие криволинейные развертки карты (максимально сохраняются расстояния между узлами в исходном пространстве).

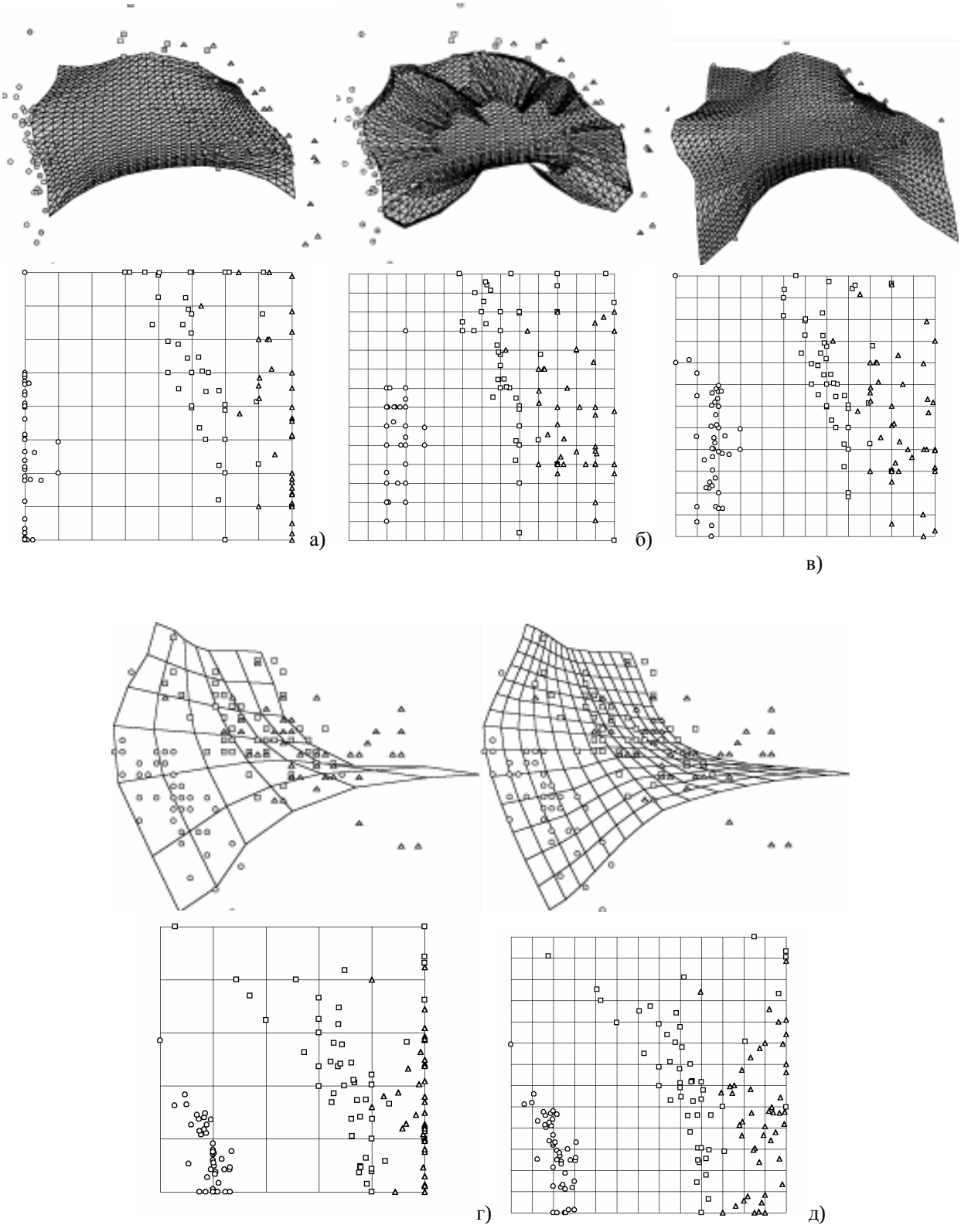


Рис. 31. Экстраполяция карты на окрестность и интерполяция карты

а) исходно построенная карта располагается «внутри» облака данных, граничные точки проецируются на крайние ребра;
б) экстраполяция по двумерной формуле Карлемана с граничным условием – узлы сходятся к центру облака данных;
в) линейная экстраполяция;
г), д) пример нелинейной интерполяции г) – исходная карта, д) – после применения процедуры интеполяции, карта в результате имеет лучшее разрешение.