Глава 4

Ассоциация образцов

Задача. Описание ассоциативной памяти.

Цели. Вы должны понять:

- разницу между автоассоциативной и гетероассоциативной сетями;
- принципы построения дискретной сети Хопфилда и двунаправленной автоассоциативной сети, чтобы иметь возможность реализовать эти сети с помощью программы электронных таблиц или на том языке программирования, который вы предпочтете;
- как использовать для автоассоциации сеть с обратным распространением ошибок;
- как обеспечить сжатие данных с помощью автоассоциативной сети с обратным распространением ошибок.

Требования. Знание основ линейной алгебры на уровне, представленном в приложении А. Знакомство с материалом главы 1.

4.1. Введение

Все знают, что такое ассоциация: слово одного языка может ассоциироваться со словом другого языка, фотография друга ассоциируется с его именем и можно даже умудриться ассоциировать некоторое туманное изображение с определенным реальным объектом. В главе 2 уже была рассмотрена идея ассоциативного обучения в виде, когда для каждого учебного образца имелся желаемый выходной образец. В этой главе мы рассмотрим случай запоминаемых пар образцов. Идея заключается в том, чтобы выбрать нужный образец из памяти, даже если у нас нет всей необходимой информации для начала поиска сохраненного образца. Например, вы хотите найти книгу в библиотеке, но не помните ее названия. При этом если вы знаете имя автора и описание того, чему книга посвящена, этого уже достаточно (с большой долей уверенности!), чтобы найти ассоциируемый с этой информацией объект.

Когда сохраняемая в памяти пара ассоциируемых образцов создается одинаковыми образцами, память называется *автоассоциативной*, а если образцы являются разными, то память называется *гетероассоциативной*. В этой главе будут рассмотрены три модели нейронных сетей для автоассоциации образцов.

4.2. Дискретная сеть Хопфилда

Сеть Хопфилда (Hopfield) является автоассоциативной сетью, ведущей себя подобно памяти, которая может вспомнить сохраненный образец даже по подсказке (в виде вводимых данных), представляющей собой искаженную помехами версию нужного образца. Например, сеть может сохранить набор изображений букв, а когда сети будет представлена искаженная версия сохраненного символа, сеть должна оказаться способной найти истинный экземпляр. Дискретная сеть Хопфилда имеет следующие характеристики.

- Один слой элементов (входные элементы, представляющие входной образец, не учитываются).
- Каждый элемент связывается со всеми другими элементами, но элемент не связывается с самим собой.
- За один шаг обновляется только один элемент, в отличие, например, от сети с обратным распространением ошибок, где все элементы слоя могут изменяться одновременно, если сеть реализована в виде аппаратных средств с соответствующими параллельными возможностями.
- Элементы обновляются в случайном порядке, но в среднем каждый элемент должен обновляться в одной и той же мере. Например, в случае сети из 10 элементов после 100 обновлений каждый элемент должен обновиться приблизительно 10 раз.
- Вывод элемента ограничен значениями 0 или 1.

Сеть Хопфилда является рекуррентной в том смысле, что для каждого входного образца выход сети повторно используется в качестве ввода до тех пор, пока не будет достигнуто устойчивое состояние. Пример сети Хопфилда показан на рис. 4.1. Удобно считать, что сеть Хопфилда не имеет входных элементов, так как входной вектор просто определяет начальные значения активности элементов. Например, если ввод является двоичным, то входной вектор [1 1 0 1] означает, что значения активности для элементов {1, 2, 4} будут равны 1, а для элемента {3} активность будет равна 0. Элемент обновляется тогда, когда все элементы передадут свои значения активности по имеющимся взвешенным связям, после чего вычисляется сумма произведений (т.е. берется скалярное произведение). Значение активности элемента получается на основе использования некоторого правила активизации. Каждый элемент сети Хопфилда имеет состояние, характеризующееся значением активности, которое должен посылать данный элемент другим элементам, а состояние сети в любой момент времени задается вектором состояний всех ее элементов.

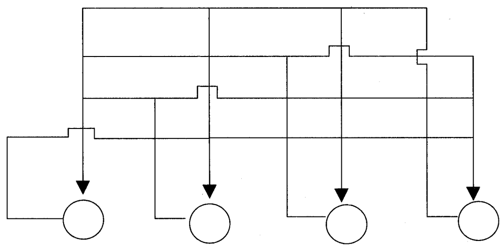

Рис. 4.1. Сеть Хопфилда с четырьмя элементами. Для каждого элемента входного вектора имеется свой элемент сети. Элементы сети связаны со всеми остальными ее элементами, но не сами с собой. Связи являются двунаправленными

В качестве входных данных сети Хопфилда можно использовать двоичные, но здесь мы будем использовать +1 для обозначения состояния "включено" и –1 — для состояния "выключено". Комбинированный ввод элемента вычисляется по формуле:

$$net_i = \sum_{i=1}^{n} s_i w_{ij} ,$$

где s_i обозначает состояние элемента с номером i. Когда элемент обновляется, его состояние изменяется в соответствии с правилом

$$s_j = \begin{cases} +1, & \text{если } net_j > 0, \\ -1, & \text{если } net_j < 0. \end{cases}$$

Эта зависимость называется сигнум-функцией и в более краткой форме она записывается в виде

$$s_j = \operatorname{sgn}(net_j) .$$

Если комбинированный ввод оказывается равным нулю, то элемент остается в состоянии, в котором он пребывал перед обновлением.

Сеть работает очень просто. Входной вектор задает начальные состояния всех элементов. Элемент для обновления выбирается случайным образом. Выбранный элемент получает взвешенные сигналы от всех остальных элементов и изменяет свое состояние. Выбирается другой элемент, и процесс повторяется. Сеть достигает предела, когда ни один из ее элементов, будучи выбранным для обновления, не меняет своего состояния.

Весовые значения для сети Хопфилда определяются непосредственно из учебных данных без необходимости проведения обучения в более привычном смысле. Сеть Хопфилда ведет себя как память, и процедура сохранения отдельного вектора представляет собой вычисление прямого произведения вектора с ним самим. В результате этой процедуры создается матрица, задающая весовые значения для сети Хопфилда, в которой все диагональные элементы должны быть установлены равными нулю (поскольку диагональные элементы задают автосвязи элементов, а элементы сами с собой не связаны). Таким образом, весовая матрица, соответствующая сохранению вектора **x**, задается формулой

$$\mathbf{W} = \mathbf{x}^{\mathrm{T}}\mathbf{x}.$$

Пример 4.1.

Найдите набор весовых значений сети Хопфилда, соответствующий сохранению образца

[1 –1 1 1].

Решение 4.1.

$$\begin{bmatrix} 1 \\ -1 \\ 1 \\ 1 \end{bmatrix} \begin{bmatrix} 1 & -1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} 1 & -1 & 1 & 1 \\ -1 & 1 & -1 & -1 \\ 1 & -1 & 1 & 1 \\ 1 & -1 & 1 & 1 \end{bmatrix}.$$

Поэтому весовыми значениями будут

$$\mathbf{W} = \begin{bmatrix} 0 & -1 & 1 & 1 \\ -1 & 0 & -1 & -1 \\ 1 & -1 & 0 & 1 \\ 1 & -1 & 1 & 0 \end{bmatrix}.$$

Первый столбец представляет весовые значения, связанные с первым элементом, столбец 2 представляет весовые значения, связанные со вторым элементом, и т.д. Если сети будет предложен образец [1 –1 1 1], то все элементы после обновления останутся в том же состоянии. Данные подсказки определяют начальные состояния всех элементов, так что в нашем случае второй элемент должен сначала находиться в состоянии –1, а все остальные — в состоянии 1. Первый элемент обновляется с помощью умножения вектора подсказки на первый столбец матрицы весов:

$$[1 \quad -1 \quad 1 \quad 1]\begin{bmatrix} 0 \\ -1 \\ 1 \\ 1 \end{bmatrix} = 3\,, \qquad \operatorname{sgn}(3) = 1.$$

Так что первый элемент останется в том же состоянии. Точно также при обновлении оставались бы неизменными и состояния всех остальных элементов.

Пример 4.2.

Найдите устойчивое состояние сети Хопфилда из примера 4.1 при условии, что входным образцом является

$$[-1\ -1\ 1\ 1].$$

Решение 4.2.

Элементы должны обновляться в случайном порядке. Для иллюстрации будем обновлять элементы в порядке 3, 4, 1, 2. Сначала рассмотрим элемент 3:

$$[-1 \quad -1 \quad 1 \quad 1]\begin{bmatrix} 1 \\ -1 \\ 0 \\ 1 \end{bmatrix} = 1\,, \qquad \operatorname{sgn}(1) = 1.$$

Таким образом, элемент 3 остается в том же состоянии. Следующим является элемент 4:

$$[-1 \quad -1 \quad 1 \quad 1]\begin{bmatrix} 1 \\ -1 \\ 1 \\ 0 \end{bmatrix} = 1\,, \qquad \operatorname{sgn}(1) = 1.$$

Так что элемент 4 остается в том же состоянии. Теперь элемент 1:

$$[-1 \quad -1 \quad 1 \quad 1]\begin{bmatrix} 0 \\ -1 \\ 1 \\ 1 \end{bmatrix} = 3\,, \qquad \operatorname{sgn}(3) = 1.$$

Так что элемент 1 изменит свое состояние с –1 на 1. Наконец, элемент 2:

$$[1 \;\; -1 \;\; 1 \;\; 1]\begin{bmatrix} -1 \\ 0 \\ -1 \\ -1 \end{bmatrix} = -3, \qquad \operatorname{sgn}(-3) = -1.$$

Так что элемент 2 останется в том же состоянии. Мы видим, что выявился ранее сохраненный вектор, характеризующий устойчивое состояние сети. Чтобы убедиться в том, что это состояние на самом деле является устойчивым, необходимо проверить, что в результате обновления ни один из элементов действительно не изменит своего состояния.

Процедура сохранения нескольких образцов в сети Хопфилда тоже проста: прямое произведение вычисляется для каждого вектора, и все полученные таким образом весовые матрицы складываются.

Пример 4.3.

Определите весовую матрицу сети Хопфилда, соответствующую сохранению следующих двух векторов

$$[-1 \;\; 1 \;\; -1],$$
$$[1 \;\; -1 \;\; 1].$$

Решение 4.3.

Соответствующей весовой матрицей является матрица

$$\begin{aligned}
\mathbf{W} &= \begin{bmatrix} -1 \\ 1 \\ -1 \end{bmatrix}[-1 \;\; 1 \;\; -1] + \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}[1 \;\; -1 \;\; 1] \\
&= \begin{bmatrix} 1 & -1 & 1 \\ -1 & 1 & -1 \\ 1 & -1 & 1 \end{bmatrix} + \begin{bmatrix} 1 & -1 & 1 \\ -1 & 1 & -1 \\ 1 & -1 & 1 \end{bmatrix} \\
&= \begin{bmatrix} 0 & -2 & 2 \\ -2 & 0 & -2 \\ 2 & -2 & 0 \end{bmatrix}.
\end{aligned}$$

Диагональные элементы были обнулены.

Эксперимент

Сеть Хопфилда использовалась для того, чтобы сохранить три образца, изображения которых показаны на рис. 4.2. Каждый образец рассматривался в виде строки биполярных значений, где 1 представляла черный цвет, а –1 — серый.

Строка состояла из 63 элементов, поэтому сеть имела 63 элемента и $63 \times 63 - 63$ весовых значения (это число весовых значений прямого произведения минус нулевые диагональные элементы). После определения весовых значений сети была предъявлена искаженная помехами версия числа "1". Образец, который выявился после некоторого числа итераций, показан на рис. 4.3.

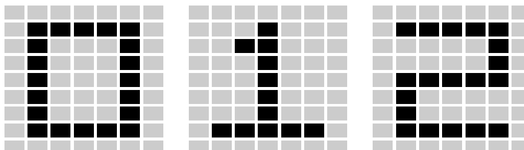

Рис. 4.2. Три образца, сохраняемые сетью Хопфилда. Черным пикселям соответствуют значения 1, а серым — значения –1

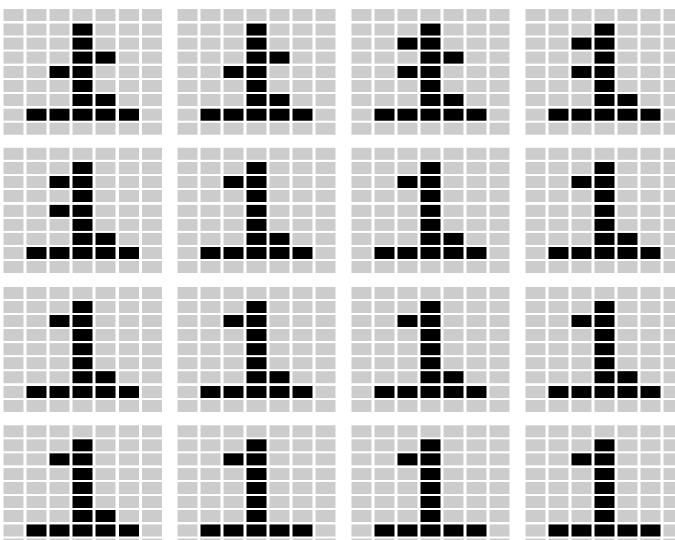

Рис. 4.3. Соответствующий вводимой подсказке образец показан в первом кадре в верхнем левом углу. Кадры представляют десять итераций сети (т.е. десять случайных обновлений). Кадры следует рассматривать в порядке слева направо и сверху вниз

Этот эксперимент демонстрирует правильное восстановление сохраненного образца по искаженной его версии. Иногда сеть Хопфилда допускает ошибки, например, выявляя неверный образец или показывая искаженную версию правильного образца (в последнем случае выявленный образец может быть распознан, но все еще искажен).

В ряде работ имеются оценки, касающиеся числа образцов, которые можно сохранить в сети Хопфилда. Одной такой общей оценкой из [Haykin, 1994] является оценка

$$p_{\max} = \frac{N}{2\ln N},$$

характеризующая максимальное число образцов, которые могут быть сохранены, если требовать правильного распознавания большинства образцов; N обозначает число элементов сети.

4.2.1. Функция энергии

В работе [Hopfield, 1984] Хопфилд доказал, что его сеть должна сходиться к устойчивому набору значений активности, рассмотрев функцию энергии системы. В представленной нами форме сеть определяет необходимость изменения состояния элемента по пороговому значению, равному нулю, поэтому мы можем использовать упрощенную версию функции энергии Хопфилда, подобную той, которая приводится в [Haykin, 1994]:

$$E = -\frac{1}{2}\sum_j \sum_i s_j s_i w_{ij} .$$

Если элемент j изменяет свое состояние на Δs_j, то изменение энергии будет равно

$$\Delta E = -\Delta s_j \sum_i s_i w_{ij} .$$

Можно рассматривать изменение энергии как функцию Δs_j и $\sum s_i w_{ij}$. Из табл. 4.1 видно, что Δs_j и $\sum s_i w_{ij}$ имеют одинаковые знаки, поэтому энергия при переходе от итерации к итерации всегда уменьшается. Например, первая строка данных таблицы говорит о том, что если элемент находится в положительном состоянии и его комбинированный ввод больше нуля (т.е. положителен), то новое состояние останется положительным и изменение состояния тоже будет положительным, а изменение энергии таким образом будет отрицательным.

Таблица 4.1. Изменения энергии при изменениях состояния элемента

Старое s_j	$\sum s_i w_{ij}$	*Новое* s_j	Δs_j	*Изменение энергии*
Положительное	Положительное	Положительное	Положительное	Отрицательное
Положительное	Отрицательное	Отрицательное	Отрицательное	Отрицательное
Отрицательное	Отрицательное	Отрицательное	Отрицательное	Отрицательное
Отрицательное	Положительное	Положительное	Положительное	Отрицательное

4.3. Двунаправленная ассоциативная память

Сетью, имеющей много общего с сетью Хопфилда, является двунаправленная ассоциативная память (сеть ВАМ — Bidirectional Associative Memory), предложенная Коско (см. [Kosko, 1988]). Сеть ВАМ является гетероассоциативной рекуррентной сетью. Сеть сохраняет пары образцов и может восстановить образец, когда ассоциированный с ним образец предлагается ей в качестве подсказки. В этой сети два слоя элементов — по одному для каждого из образцов пары — и оба слоя соединяются двунаправленными связями (т.е. активность может передаваться по связям в обоих направлениях) (рис. 4.4).

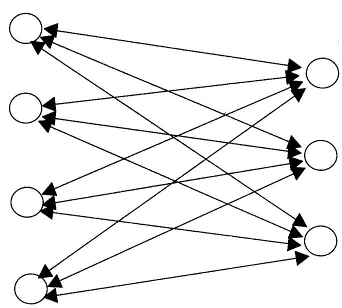

Рис. 4.4. Двунаправленная ассоциативная память. Элементы слева представляют образцы размерности 4, а элементы справа — ассоциированные с ними образцы размерности 3

Здесь мы рассмотрим только дискретную биполярную сеть ВАМ, но можно рассмотреть и непрерывные значения. Чтобы сохранить образец **s** и ассоциируемый с ним образец **t**, рассматривается прямое произведение, определяющее весовые значения. Процедура точно такая же, как и в сети Хопфилда, но теперь матрица уже не обязана быть квадратной, а диагональные элементы не обнуляются. Весовой матрицей для одной пары будет матрица

$$\mathbf{W} = \mathbf{s}^{\mathrm{T}}\mathbf{t}\,.$$

Чтобы сохранить несколько пар, все соответствующие произведения, определяющие весовые значения, складываются, точно так же, как это делается для сети Хопфилда.

Процедура нахождения в памяти элемента подобна соответствующей процедуре сети Хопфилда. В следующем описании i обозначает один слой элементов, а j — ассоциированный слой элементов.

- Устанавливаются значения активности элементов слоя i в соответствии со значениями, задаваемыми входным образцом.
- Распространяется активность на слой j. Комбинированный ввод элемента слоя j равен

$$net_j = \sum_i s_i w_{ij}\,.$$

- Вычисляется новое состояние для каждого элемента слоя j:

$$t_j = f(net_j)\,.$$

- Распространяется активность на слой i. Комбинированный ввод элемента слоя i равен:

$$net_i = \sum_j s_j w_{ji}\,.$$

- Вычисляется новое состояние для каждого элемента слоя i:

$$s_i = f(net_i)\,.$$

- Это двустороннее распространение сигналов активности повторяется до тех пор, пока не будет достигнуто устойчивое состояние. Активность для каждого слоя определяется относительно некоторой пороговой величины θ.

$$t_j = f(net_j) = \begin{cases} 1, & \text{если } net_j > \theta_j, \\ t_j, & \text{если } net_j = \theta_j, \\ -1, & \text{если } net_j < \theta_j, \end{cases}$$

$$s_i = f(net_i) = \begin{cases} 1, & \text{если } net_i > \theta_i, \\ s_i, & \text{если } net_i = \theta_i, \\ -1, & \text{если } net_i < \theta_i. \end{cases}$$

Все элементы сети сначала имеют нулевые значения активности. Обратите внимание на то, что распространение может начаться с любого уровня, так как и **s** может использоваться для вызова **t**, и, наоборот, — **t** может использоваться для вызова **s**.

Пример 4.4.

1. На рис. 4.5 показаны три образца (изображения цифр 1, 2 и 3). Эти три образца должны быть сохранены бипо-лярной сетью ВАМ. Ассоциируемыми с ними образцами являются трехбитовые двоичные числа (конвертированные в биполярную форму). Ассоциированные образцы представлены в табл. 4.2. Предложите весовые значения для этой сети.
2. Покажите, что каждая ассоциация может быть вызвана из памяти.

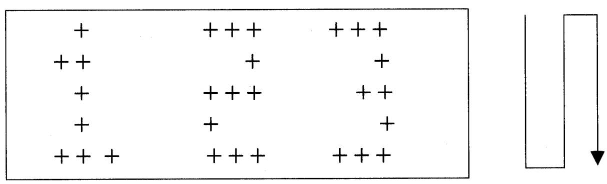

Рис. 4.5. Каждая цифра изображена на сетке 5 × 3. Пробел кодируется значением –1, а знак '+' — значением +1. Цифры представляются линейным массивом значений, получаемым при движении по сетке сверху вниз и слева направо

Таблица 4.2. Цифры {1,2,3}, ассоциированные с биполярными образцами

Образец	*Ассоциированный образец*
1	–1 –1 1
2	–1 1 –1
3	–1 1 1

Решение 4.4.

1.

$$
\mathbf{W} = \begin{bmatrix} -1 \\ 1 \\ -1 \\ -1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ -1 \\ -1 \\ -1 \\ -1 \\ 1 \end{bmatrix} [-1 \quad -1 \quad 1] + \begin{bmatrix} 1 \\ -1 \\ 1 \\ 1 \\ 1 \\ 1 \\ -1 \\ 1 \\ -1 \\ 1 \\ 1 \\ 1 \\ 1 \\ -1 \\ 1 \end{bmatrix} [-1 \quad 1 \quad -1] + \begin{bmatrix} 1 \\ -1 \\ -1 \\ -1 \\ 1 \\ 1 \\ -1 \\ 1 \\ -1 \\ 1 \\ -1 \\ -1 \\ -1 \\ -1 \\ 1 \end{bmatrix} [-1 \quad 1 \quad 1],
$$

$$
\therefore \ \mathbf{W} = \begin{bmatrix} -1 & 3 & -1 \\ 1 & -3 & 1 \\ 1 & 1 & -3 \\ 1 & 1 & -3 \\ -3 & 1 & 1 \\ -3 & 1 & 1 \\ 1 & -3 & 1 \\ -3 & 1 & 1 \\ 1 & -3 & 1 \\ -3 & 1 & 1 \\ -1 & 3 & -1 \\ -1 & 3 & -1 \\ -1 & 3 & -1 \\ 1 & 1 & 1 \\ -3 & 1 & 1 \end{bmatrix}
$$

2. Сначала в качестве входного рассмотрим образец, представляющий оцифрованное изображение цифры “2”:

комбинированный ввод для слоя j

$$= [1\ \ -1\ \ 1\ \ 1\ \ 1\ \ 1\ \ -1\ \ 1\ \ -1\ \ 1\ \ 1\ \ 1\ \ 1\ \ -1\ \ 1] \times \begin{bmatrix} -1 & 3 & -1 \\ 1 & -3 & 1 \\ 1 & 1 & -3 \\ 1 & 1 & -3 \\ -3 & 1 & 1 \\ -3 & 1 & 1 \\ 1 & -3 & 1 \\ -3 & 1 & 1 \\ 1 & -3 & 1 \\ -3 & 1 & 1 \\ -1 & 3 & -1 \\ -1 & 3 & -1 \\ -1 & 3 & -1 \\ 1 & 1 & 1 \\ -3 & 1 & 1 \end{bmatrix}$$

$$= [-21\ \ 27\ \ -9].$$

С учетом порогового значения получаем [–1 1 –1], что и будет ассоциируемым образцом.

Образец [–1 1 –1] можно использовать в качестве входного:

комбинированный ввод для слоя j

$$= [-1\ \ 1\ \ -1] \times \begin{bmatrix} -1 & 3 & -1 \\ 1 & -3 & 1 \\ 1 & 1 & -3 \\ 1 & 1 & -3 \\ -3 & 1 & 1 \\ -3 & 1 & 1 \\ 1 & -3 & 1 \\ -3 & 1 & 1 \\ 1 & -3 & 1 \\ -3 & 1 & 1 \\ -1 & 3 & -1 \\ -1 & 3 & -1 \\ -1 & 3 & -1 \\ 1 & 1 & 1 \\ -3 & 1 & 1 \end{bmatrix}^T = \begin{bmatrix} 5 \\ -5 \\ 3 \\ 3 \\ 3 \\ 3 \\ -5 \\ 3 \\ -5 \\ 3 \\ 5 \\ 5 \\ 5 \\ -1 \\ 3 \end{bmatrix}^T.$$

Оригинальный соответствующий изображению образец воссоздается после прохождения полученного вектора через функцию активности.

Если процесс повторить для других векторов, будет видно, что вызываются все ассоциации.

4.4. Автоассоциативное обратное распространение ошибок

Стандартная сеть с прямой связью и обратным распространением ошибок, рассмотренная в главе 2, может быть обучена автоассоциативным методом выполнению задач типа сжатия изображений. Этот метод дает возможность сделать целевой образец таким же, как и учебный образец, так что сеть учится воспроизводить в выходном слое то, что подается ей на рассмотрение во входном слое. Базовая архитектура сети показана на рис. 4.6. Сеть обучается стандартным образом, но целью обучения является ассоциация каждого учебного образца с самим собой. После успешного окончания обучения сеть может работать как два механизма: первый слой весов может обеспечивать сжатие образца, а второй слой весов может восстанавливать полный образец из его сжатого представления. Конечно, в результате реконструкции могут быть и потери информации, поскольку сеть не будет в совершенстве создавать на выходном уровне все учебные экземпляры. Образец сжимается после его подачи на вход сети и распространения сигналов, по которым вычисляются значения активности скрытых элементов. Для сети с архитектурой типа

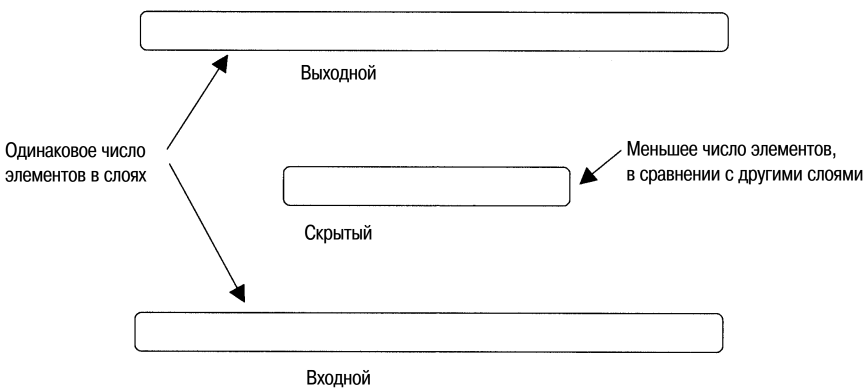

Рис. 4.6. Автоассоциативная сеть с прямой связью

20-10-20 (20 входных элементов, 10 скрытых элементов и 20 выходных элементов) отношением сжатия будет 2-1, поскольку для данного входного вектора сжатый вектор задается значениями активности скрытых элементов. Чтобы восстановить входной вектор, сжатый вектор предъявляется скрытому слою, и активность распространяется на выходной слой.

Иногда может оказаться выгодным обработать данные с помощью автоассоциативной сети перед тем, как рассмотреть их с помощью другой сетевой модели (и даже перед тем, как выполнить их классификацию с помощью другой сети с прямой связью). Известно, например, что автоассоциативная сеть с прямой связью, имеющая один скрытый слой, вычисляет по сути главные компоненты. Анализ главных компонент является основным методом статистической обработки данных и используется для того, чтобы удалить избыточность данных и провести кластеризацию. В результате анализа главных компонент происходит также декорреляция признаков векторов (признаки характеризуют положение элементов), что иногда оказывается полезным при обучении другой сети с прямой связью. Идея заключается в преобразовании учебных данных в некоторое сокращенное описание, где компонентами такого сокращенного описания выступают скрытые элементы.

Анализ главных компонент можно использовать для выполнения сжатия с потерями. Например, на рис. 4.7 показана некоторая совокупность двумерных данных. Через эту совокупность данных проведена новая пара осей, пересекающихся под прямым углом. Точки данных теперь могут быть выражены в координатах этих новых "главных осей". Более длинная ось указывает направление наибольшего разброса или дисперсии данных. Если дисперсия в направлении более длинной главной оси составляет 85% разброса данных, а в направлении второй оси — 15%, то 85% дисперсии данных сохранится, даже если вторую главную координату из рассмотрения исключить. Другими словами, если данные описать только в терминах наибольшего главного компонента, то в основном информация о данных сохранится, но некоторая ее часть все же будет потеряна. Иногда в данных присутствует некоторая избыточность, как показано на рис. 4.8. Точки данных размещаются на прямой $x_2 = x_1$. Любая точка данных может быть выбрана в качестве единственной "главной координаты" без какой бы то ни было потери информации. Поэтому если данные выразить через такую главную координату, то каждый образец будет описан одним числом вместо двух. На первый взгляд, то, что образцы могут быть описаны одной координатой вместо двух (но только в случае, когда имеется избыточность, не забывайте!), может показаться странным. Ознакомьтесь тогда с табл. 4.3, в которой набор двумерных точек описывается с помощью представления их значением одной координаты. Трансформация двумерных образцов в одномерные происходит следующим образом:

$$\begin{bmatrix} 1 & 1 \\ 3 & 3 \\ 4 & 4 \\ 8 & 8 \end{bmatrix} \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix} = [1.414 \quad 4.243 \quad 5.657 \quad 11.314].$$

Таблица 4.3. Примеры представления точек прямой $x_2 = x_1$ значениями одной координаты

x1	*x2*	*Одно значение координаты*
1	1	1 414
3	3	4.243
4	4	5.657
8	8	11.314

Обратное отображение для 1.414 задается с помощью следующего выражения:

$$[1.414] \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix} = [1 \quad 1].$$

Точно так же могут быть восстановлены и другие координаты.

Рис. 4.7. Анализ главных компонент может использоваться для представления данных с помощью нового набора координатных осей

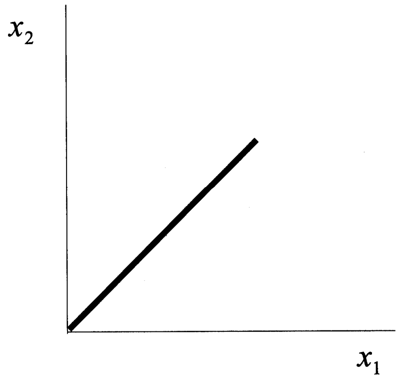

Рис. 4.8. Эта прямая состоит из точек, описываемых двумя координатными значениями, но может быть представлена и с помощью одной координаты (т.е. в одномерном виде)

Более высокого показателя сжатия можно иногда достичь с помощью дополнительных скрытых слоев. Например, сеть с архитектурой 40-20-8-20-40 может обеспечить отношение сжатия 40 к 8, если, конечно, ее обучение пройдет успешно.

4.5. Резюме

Ассоциативная сеть действует подобно памяти.

- Автоассоциация связывает образец с ним самим. Например, автоассоциативная память может использоваться для того, чтобы восстановить истинную версию сохраненного образца по искаженной версии этого образца.
- Гетероассоциация представляет собой связывание двух разных образцов. Один образец может использоваться в качестве подсказки, по которой из памяти извлекается другой образец.
- Когда автоассоциативное обучение происходит путем пропускания сигналов через узкий канал типа скрытого слоя автоассоциативной сети с прямой связью, происходит в некотором смысле сжатие данных. Сжатие может оказаться полезной формой предварительной обработки данных перед использованием их для обучения сети с другой архитектурой.

- Сеть Хопфилда может использоваться для автоассоциации, а двунаправленная ассоциативная память — для гетероассоциации.
- Весовые значения дискретной сети Хопфилда и двунаправленной ассоциативной памяти могут быть найдены с помощью простых матричных вычислений, поэтому такие сети не требуют длительного обучения.
- От числа элементов в ассоциативной сети зависит число образцов, которые могут быть сохранены.

4.6. Дополнительная литература

Подробное обсуждение сети Хопфилда и автоассоциативной сети с обратным распространением ошибок имеется в книге [Haykin, 1994].

4.7. Упражнения

1. Определите весовые значения сети Хопфилда, соответствующие сохранению образца [1 1 1 –1].

2. Для сети из упражнения 1 рассмотрите подсказку в виде входного образца [1 1 1 –1] и проверьте, стабилизируется ли сеть на сохраненном образце.

3. Повторите вычисления упражнения 2 для образца [–1 –1 1 –1].

4. Сколько образцов можно вызвать из сети Хопфилда, если каждый сохраненный вектор имеет 10 элементов?

5. (а) Определите весовые значения сети Хопфилда для сохранения следующих образцов:

$$[-1\ \ 1\ \ -1\ \ -1\ \ 1\ \ -1\ \ -1\ \ -1],$$
$$[-1\ \ -1\ \ 1\ \ -1\ \ -1\ \ -1\ \ 1\ \ -1].$$

(б) Проверьте устойчивость сети при предоставлении ей на вход в качестве подсказок сохраненных образцов.

(в) Проверьте устойчивость сети при предоставлении ей на вход в качестве подсказок следующих образцов:

$$[-1\ \ 1\ \ -1\ \ -1\ \ 1\ \ -1\ \ -1\ \ 1],$$
$$[-1\ \ 1\ \ 1\ \ -1\ \ 1\ \ -1\ \ -1\ \ -1].$$

6. Определите весовые значения сети Хопфилда для сохранения следующих образцов:

$$[-1\ \ 1\ \ -1\ \ -1\ \ 1\ \ -1\ \ -1\ \ -1],$$
$$[-1\ \ -1\ \ 1\ \ -1\ \ -1\ \ -1\ \ 1\ \ -1],$$
$$[1\ \ -1\ \ -1\ \ 1\ \ -1\ \ 1\ \ -1\ \ -1].$$

Проверьте устойчивость сети при предоставлении ей на вход в качестве подсказки первого сохраненного образца.

7. Определите сеть BAM для примера 4.4, но в качестве ассоциированных образцов для изображений цифр возьмите следующие векторы.

Образец	*Ассоциированный образец*
1	–1 1
2	1 –1
3	1 1

Сравните полученные результаты с результатами, представленными в описании примера 4.4.