

Подготовка данных при разработке медицинских экспертных систем

Розглянуто етапи розробки медичних експертних систем, та методи підготовки даних для них. Розглянуто і реалізовано методи визначення інформативної інформації із складу факторів ризику на основі нейронних мереж та генетичних алгоритмів. Розроблена архітектура нейронної мережі, підібрані методи генетичних операторів, розроблена фітнес-функція. Проведені випробування і наведені результати використання методів на реальних медичних даних.

Введение. Предвидение явлений в медицине является наиболее актуальной научно-практической задачей профессиональной деятельности врача. Совершенствование известных и создание новых методов диагностики и прогнозирования позволяет существенно улучшить качество оказания медицинской помощи населению. Задачи диагностики, прогнозирования и принятия решений в медицине – это комплексный процесс, который охватывает шаги, начиная от получения и представления данных до оценки качества полученных решений. В целом весь процесс можно разделить на следующие этапы:

- отбор данных;
- предобработка данных;
- редукция данных;
- поиск закономерностей;
- оценка и интерпретация найденных закономерностей;
- использование полученных знаний для поставленной задачи.

В данной работе рассматриваются задачи подготовки (предобработки и редукции) данных для построения медицинских экспертных систем. При этом предполагается, что отбор данных выполняется врачом. Врач, при постановке диагноза, всегда старается анализировать весь комплекс сведений о пациенте. Эта информация имеет самый разнообразный вид: термины, описывающие какие-либо заболевания, числа различного вида и величины и т.д. При этом одна часть этих параметров имеет принципиальное значение для принятия решения, другая же не столь важна и даже может просто игнорироваться. Достаточно часто для диагностики и прогнозирования больному проводят сложные и дорогостоящие методы обследования, порой небезвредные для здоровья. Однако во многих случаях представляется возможным получить ответ и без них. Как правило, в медицинских задачах, нет никакой дополнительной информации о том, какие данные действительно нужны для решения поставленной задачи. Анализ всех факторов риска обычно вызывает существенные затруднения при построении и обучении экспертных систем. Поэтому разработка представления обучающих данных - очень важный этап, который в значительной степени определяет качество получаемой экспертной системы.

Постановка задачи. Существует огромное количество способов представления информации для различных целей. Конечно в первую очередь, выбор зависит от метода разработки экспертной системы [1]. Но, в любом случае, экспертная система оперирует с информацией, представленной только в виде чисел. Числа подаются на входы экспертной системы и ответы, снимаемые с выходов, также представляют собой числа. Поэтому возникает необходимость корректного представления входной информации в виде чисел, сохраняющих смысл и внутренние взаимосвязи в данных. Следующая задача, которую необходимо решить – это сократить размерность входных данных [2,3,4].

Рассмотрим типы данных, с которыми столкнулись при создании медицинских экспертных систем, а так же оптимальные способы их представления в численном виде.

Числовые признаки. Числовые значения могут быть совершенно разнородными величинами. Очевидно, что результаты, например, нейросетевого моделирования не должны зависеть от единиц измерения этих величин. А именно, чтобы сеть трактовала эти значения единообразно, все входные и выходные величины должны быть приведены к единому (как правило, единичному) масштабу. Кроме того, для повышения скорости и качества обучения полезно провести дополнительную предобработку данных, выравнивающую распределение значений еще до этапа обучения.

Приведение к единому масштабу обеспечивается нормировкой каждой переменной на диапазон разброса ее значений. В простейшем варианте это – линейное преобразование: в единичный отрезок: $\tilde{x}_i \in [0,1]$.

$$\tilde{x}_i = \frac{x_i - x_{i,\min}}{x_{i,\max} - x_{i,\min}} \quad (1)$$

Обобщение для отображения данных в интервал $[-1,1]$, рекомендуемого для входных данных тривиально.

Линейная нормировка оптимальна, когда значения переменной x_i плотно заполняют определенный интервал. Но подобный «прямолинейный» подход применим далеко не всегда. Так, если в данных имеются относительно редкие выбросы, намного превышающие типичный разброс, именно эти выбросы определяют согласно формуле (1) масштаб нормировки. Это приводит к тому, что основная масса значений нормированной переменной \tilde{x}_i сосредотачивается вблизи нуля $|\tilde{x}_i| \ll 1$. Гораздо надежнее, поэтому, ориентироваться при нормировке не на экстремальные значения, а на типичные, т.е. статистические характеристики данных, такие как среднее и дисперсия.

$$\tilde{x}_i = \frac{x_i - \bar{x}_i}{\sigma_i} \quad (2)$$

$$\text{где } \bar{x}_i \equiv \frac{1}{P} \sum_{\alpha=1}^P x_i^\alpha, \quad \sigma_i^2 \equiv \frac{1}{P-1} \sum_{\alpha=1}^P (x_i^\alpha - \bar{x}_i)^2 \quad (3) \text{ и } (4)$$

В этом случае основная масса данных будет иметь единичный масштаб, т.е. типичные значения все переменных будут сравнимы. Но теперь нормированные величины не принадлежат гарантированно единичному интервалу, более того, максимальный разброс значений \tilde{x}_i заранее не известен. Для входных данных это может быть и не важно, но выходные переменные будут использоваться в качестве эталонов для выходных нейронов. В случае, если выходные нейроны – сигмоидные, они могут принимать значения лишь в единичном диапазоне. Чтобы установить соответствие между обучающей выборкой и нейросетью в этом случае необходимо ограничивать диапазон изменения переменных.

Линейное преобразование, представленное выше, не способно произвести нормировку основной массы данных и одновременно ограничить диапазон возможных значений этих данных. Естественный выход из этой ситуации – использовать для предобработки данных функцию активации используемых нейронов. Например, нелинейное преобразование нормирует основную массу данных, одновременно гарантируя что $\tilde{x}_i \in [0,1]$

$$\tilde{x}_i = f\left(\frac{x_i - \bar{x}_i}{\sigma_i}\right), \quad f(x) = \frac{1}{1 + e^{-x}} \quad (5) \text{ и } (6)$$

В общем случае – это функциональная предобработка, которая отображает входной признак x в n -мерный вектор x^{\square} . Наиболее часто используются следующие варианты:

- линейная предобработка. В линейной предобработке используется кусочно линейная функция:

$$\tilde{x}_i = \begin{cases} a, & x < a, \\ x, & a \leq x \leq b, \\ b, & b < x. \end{cases} \quad (7)$$

- сигмоидная предобработка. В сигмоидной предобработке может использоваться любая сигмоидная функция.
- шапочная предобработка. Для шапочной предобработки используются любые функции, имеющие график в виде «шапочки».

Функциональная предобработка на примере логистической функции была рассмотрена выше. Формулы (5) и (6).

Бинарные признаки. Бинарные признаки характеризуются наличием только двух состояний – истина и ложь. Однако даже такие простые данные могут иметь два разных смысла. Значение истина означает наличие у описываемого объекта какого-либо свойства. А ответ ложь может означать либо отсутствие этого свойства, либо наличие другого свойства. В зависимости от смысловой нагрузки значения ложь, и учитывая заданный диапазон $[a,b]$, рекомендуемые способы кодирования бинарного признака приведены в таблице 1.

Неупорядоченные качественные признаки. Признак называют неупорядоченным, если никакие два состояния нельзя связать естественным в контексте задачи отношением порядка. Информация при этом может быть представлена в виде одного варианта из заранее известного и ограниченного набора вариантов, или одновременно несколькими вариантами из известного и ограниченного набора вариантов.

Таблица 1.

Кодирование бинарного признака

Смысл значения ложь	Значение входного сигнала	
	Истина	Ложь
Отсутствие заданного свойства при $b = 0$	a	0
Отсутствие заданного свойства при $b \neq 0$	b	0
Наличие другого свойства	b	a

Поскольку никакие два состояния признака не связаны каким-либо отношением порядка, то было бы неразумным кодировать их разными величинами одного входного сигнала нейронной сети. Поэтому, для кодирования качественных признаков рекомендуется использовать столько входных сигналов, сколько состояний у этого качественного признака. Каждый входной сигнал соответствует определенному состоянию.

Наиболее естественной выглядит и чаще всего используется на практике двоичное кодирование типа « $n \rightarrow n$ » когда имена n категорий кодируются значениями n бинарных нейронов, причем первая категория кодируется как $(1,0,0,\dots,0)$, вторая, соответственно – $(0,1,0,\dots,0)$ и т.д. вплоть до n -ной: $(0,0,0,\dots,1)$. Легко убедиться, что при такой симметричной кодировке расстояния между всеми векторами-категориями равны.

Если же входной диапазон равен $[a,b]$, кодировка может быть выполнена следующим образом. Обозначим набор всех состояний рассматриваемого признака через a_1, a_2, \dots, a_n , рекомендуемая таблица кодировки имеет вид, приведенный в таблице 2.

Таблица 2.

Кодирование неупорядоченного качественного признака

Состояние	Вектор входных сигналов
a_1	(b, a, a, \dots, a)
a_2	(a, b, a, \dots, a)
a_n	(a, a, \dots, a, b)

Если информация представлена в виде одного и только одного варианта из заранее известного и ограниченного набора вариантов, то можно использовать более компактный код $n \rightarrow m$, когда имена n классов кодируются m -битным двоичным кодом. Причем, в новой кодировке активность кодирующих нейронов должна быть равномерна: иметь приблизительно одинаковое среднее по примерам значение активации. Это гарантирует одинаковую значимость весов, соответствующих различным нейронам.

Упорядоченные качественные признаки. Упорядоченным признаком называется такой признак, для любых двух состояний которого можно сказать, что одно из них предшествует другому. Такие данные находятся в определенных взаимосвязях друг с другом, которые могут быть выражены отношениями типа "больше", "меньше". Примером может служить степень тяжести заболевания (I, II, III) или (легкий больной < средний больной < тяжелый больной). В любом случае варианты располагаются в определенном порядке - как правило, возрастания или убывания.

Один из вариантов кодирования таких переменных - поставить в соответствие номерам категорий такие числовые значения, которые сохраняли бы существующую упорядоченность. Естественно, при этом имеется большая свобода выбора - любая монотонная функция от номера класса порождает свой способ кодирования. При этом мы должны стремиться к тому, чтобы максимизировать энтропию закодированных данных. При использовании сигмоидных функций активации все выходные значения лежат в конечном интервале - обычно $[0, 1]$ или $[-1, 1]$. Из всех статистических функций распределения, определенных на конечном интервале, максимальной энтропией обладает равномерное распределение.

Применительно к данному случаю это подразумевает, что кодирование переменных числовыми значениями должно приводить, по возможности, к равномерному заполнению единичного интервала закодированными примерами, захватывая при этом и этап нормировки. При таком способе все примеры будут нести примерно одинаковую информационную нагрузку. Таким образом, можно переменные кодировать следующим образом: единичный отрезок разбивается на n отрезков (n равно числу классов), с длинами пропорциональными числу примеров каждого класса в обучающей выборке. Например, переменная, которая содержит информацию о курении может принимать значения: мало, средне, много. Пусть x_1 - мало, x_2 - средне и x_3 - много. Тогда $\Delta x_n = P_n/P$, где P_n - число примеров данного класса n , а P - общее число примеров. Центр каждого такого отрезка будет являться численным значением для соответствующего класса (Рисунок 1).

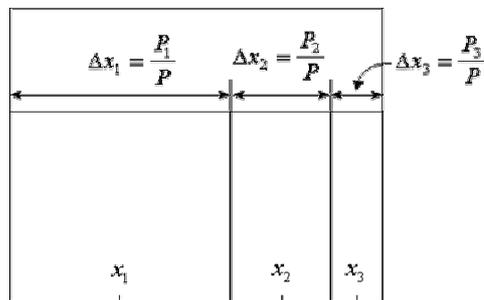


Рисунок 1. Иллюстрация способа кодирования одной переменной упорядоченного типа.

Кодирование таким способом, т.е. разными значениями одного входного сигнала не всегда эффективно из-за того, что расстояние между состояниями не определено, а такой способ кодирования эти расстояния задает явным образом. Поэтому, упорядоченные частные признаки можно, как и неупорядоченные, кодировать в виде столько входных сигналов, сколько состояний у признака. Но, в отличие от неупорядоченных признаков, накапливать число сигналов с максимальным значением. Если все состояния обозначены через $a_1 < a_2 < \dots < a_n$, рекомендуемая таблица кодировки приведена в таблице 3.

Таблица 3.

Кодирование упорядоченного качественного признака

Состояние	Вектор входных сигналов
a_1	(b, a, a, \dots, a)
a_2	(b, b, a, \dots, a)
a_n	(b, b, b, \dots, b)

Редукция данных. Существует два подхода к решению данной задачи:

1. Выделение полезных входных переменных.
2. Понижение размерности.

Методы искусственного интеллекта вычисления информативности признаков. Существует несколько методов позволяющие выявить относительную значимость входных параметров, с помощью нейронных сетей [2,5].

Для выделения полезных входных переменных с помощью нейронных сетей можно использовать так называемый метод проб и ошибок, придерживаясь одной из двух стратегий: наращивания или отсекания.

При наращивании, мы начинаем с одной переменной и по одной добавляем к ней другие переменные. Если результат от этого улучшается, то такая комбинация запоминается. Такой подход имеет существенный недостаток – игнорируется то обстоятельство, что две или более переменных могут быть взаимосвязаны (т.е. может оказаться так, что нужно добавить сразу несколько переменных, чтобы улучшить результат).

При отсекании, мы берем все имеющиеся переменные и начинаем их по одной убирать. Если результат от этого ухудшается, то возвращаемся к предыдущей комбинации переменных. Как правило, такой подход позволяет сохранить взаимосвязанные переменные, но он достаточно медленный и не рациональный.

Подобную стратегию можно реализовать с помощью генетических алгоритмов [6], они являются эффективным инструментом поиска решений в комбинаторных задачах. Схема работы генетического алгоритма такова: каждый возможный вариант набора входных переменных можно представить в виде битовой маски. Ноль в соответствующей позиции означает, что эта входная переменная не включена во входной набор, единица – что включена. Таким образом, маска представляет собой строку битов – по одному на каждую возможную входную переменную – и генетический алгоритм оптимизирует такую битовую структуру. Алгоритм следит за некоторым набором таких строк, оценивая каждую из них по контрольной ошибке (ошибка обучения). По значениям ошибки производится отбор лучших вариантов масок, которые комбинируются друг с другом с помощью искусственных генетических операций: скрещивания и мутации.

Понижение размерности. Наиболее распространенный метод понижения размерности – это анализ главных компонент (АГК). Метод состоит в следующем: к данным применяется линейное преобразование, при котором направлениям новых координатных осей соответствуют направления наибольшего разброса исходных данных. Как правило, уже первая компонента отражает большую часть информации, содержащейся в данных. Достаточно часто метод АГК выделяет из многомерных исходных данных совсем небольшое число компонент, сохраняя при этом структуру информации. Однако один из недостатков метода заключается в его линейности и, следовательно, в невозможности учесть некоторые

важные характеристики структуры данных. Применим нелинейный вариант АГК, основанный на применении автоассоциативных нейронных сетей [2].

Автоассоциативная сеть – это сеть, предназначенная для воспроизведения на выходе своих же входных данных. У такой сети число выходов совпадает с числом входов. Число скрытых элементов делается меньше числа входов/выходов, и это заставляет сеть «сжимать» информацию, представляя ее в меньшей размерности.

Трехслойная автоассоциативная сеть сначала линейно преобразует входные данные в меньшую размерность промежуточного слоя, а затем снова линейно разворачивает их в выходном слое. Такая сеть на самом деле реализует стандартный алгоритм АГК. Для того чтобы выполнить нелинейное понижение размерности, нужно использовать пятислойную сеть. Средний слой такой сети служит для уменьшения размерности, а соседние с ним слои, отделяющие от входного и выходного слоев, выполняют нелинейные преобразования.

Для осуществления нелинейного понижения размерности с помощью автоассоциативной сети необходимо:

- сформировать обучающий набор данных для автоассоциативной сети;
- построить автоассоциативную сеть с пятью слоями. В среднем скрытом слое меньше элементов, чем во входном и выходном слоях. В двух оставшихся промежуточных слоях достаточно большое (и одинаковое) число элементов;
- обучить автоассоциативную сеть на подготовленном обучающем множестве;
- удалить два последних слоя автоассоциативной сети, в результате получается сеть, способная понижать размерность.

Реализация рассмотренных методов. Для реализации поставленной задачи написана программа в среде визуального объектно-ориентированного программирования C++ Builder 6. С ее помощью можно построить и обучить нейронную сеть типа многослойный персептрон, произвести отбор входных параметров с помощью генетических алгоритмов и понизить размерность с помощью автоассоциативной сети. Архитектура сети определяется количеством слоев, количеством нейронов на каждом слое и активационной функцией для каждого слоя (здесь предусмотрены наиболее распространенные для решения подобных задач варианты: линейная, сигмоидальная, гиперболический тангенс, гладкая ступенчатая). Предусмотрена возможность использования пре- и пост-процессинга входных данных. Входные и выходные данные можно загрузить из файла Microsoft Excel.

Сеть обучается алгоритмом обратного распространения ошибки [5]. Так как в общем случае не существует доказательства сходимости алгоритма, то и не существует какого-либо четкого определенного критерия останова. Есть несколько обоснованных критериев, которые можно использовать. Каждый из них имеет свои практические преимущества. Одним из таких критериев является малая интенсивность изменений среднеквадратической ошибки в течение эпохи. Интенсивность изменения среднеквадратической ошибки обычно считается достаточно малой, если она лежит в пределах 0,1 – 1% за эпоху. Иногда используется уменьшенное значение – 0,01%. К сожалению, такой критерий может привести к преждевременной остановке процесса обучения. Другим критерием может быть достижение целевого значения среднеквадратической ошибки. Недостатком этого критерия является то, что для сходимости обучения может потребоваться довольно много времени. Так же можно задавать определенное количество эпох обучения. Учитывая, что нет информации о том, сколько их может потребоваться, данный критерий тоже не всегда удобен. Если необходимо создать сеть, обладающую хорошей способностью к обобщению, можно использовать перекрестную проверку [5]. В данном случае обучающее множество делится на подмножество для обучения и проверочное подмножество, которое используется для проверки эффективности различных моделей сети, из которых необходимо выбрать лучшую. В разработанной программе предусмотрены все перечисленные выше критерии останова.

Генетические алгоритмы используют архитектуру подобранной сети и пытаются ее обучить используя различные комбинации набора входных переменных. При этом предусмотрены следующие возможности: в качестве метода селекции можно выбирать колесо рулетки или турнир (с указанием количества особей в туре); в качестве метода редукции предусмотрена элитарная стратегия, полная замена, частичная замена популяции (с указанием процента заменяемых особей); можно задавать вероятность операции мутации и скрещивания, (предусмотрен одноточечный и двухточечный оператор скрещивания). В качестве критерия останова можно использовать определенное количество итераций или указать количество повторений результата. Предложена следующая функция приспособленности:

$$F = \left(\frac{X_i}{X_n} \right) * W_1 + \left(\frac{E_i - E_n}{E_n} \right) * W_2 \quad (8)$$

где X_i – количество единиц для i – ой хромосомы, X_n – максимальное количество единиц, E_i – ошибка обучения для i – ой хромосомы, E_n – ошибка обучения при использовании максимального количества факторов, W_1 и W_2 – мера влияния на фитнес-функцию.

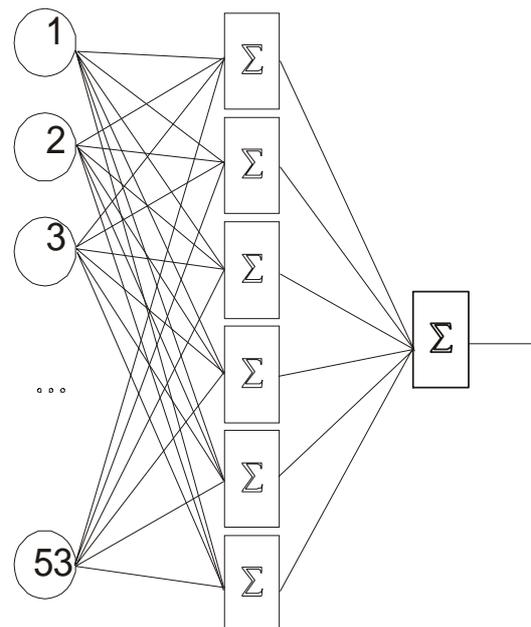
Меру влияния каждого слагаемого можно корректировать вручную. Диапазон допустимых значений - $W_1, W_2 \in (0,1)$, и должно выполняться условие: $W_1 + W_2 = 1$.

Апробация рассмотренных методов. Для тестирования использовались данные полученные при обследовании 120 детей, которые умерли за период 1990-1999г. (71 мальчик и 49 девочек) в Донецкой области от синдрома внезапной смерти грудных детей (СВСГД), и контрольная группа из 120 живых детей на первом году жизни, подобранных по принципу копий-пар в соответствии с возрастом, полом, годом и месяцем рождения, а также географическим распределением в рамках города. Собрана максимально полная информация о возможных параметрах, которые в той или иной степени могут влиять на СВСГД. К возможным факторам риска выделили следующую информацию:

- информация о матери: место жительства – город или село; вредные условия труда; образование; состоит ли в браке; бытовые условия и количество м2 на человека; рост и вес; возраст на момент первой беременности; чем закончилась первая беременность; возраст на момент первых месячных; регулярность, болезненность, длительность и интервал месячных; возраст на момент беременности; номер беременности; роды по счету; чем закончились предыдущая беременность; количество аборт, самоабортов, мертворождений; плодность текущей беременности; курение, алкоголь, наркотики в течении беременности; перенесенные заболевания; способы контрацепции; TORCH – инфекции; патология беременности; гинекологические заболевания; группа крови и резус фактор.
- информация об отце: возраст; курение; алкоголь; наркотики.
- информация о ребенке: пол, кормили грудью, искусственное питание или смешанное; вес; рост; количество баллов по шкале Апгар; срок гестации; врожденные пороки; сразу после родов находился: в палате интенсивной терапии, в палате, с мамой.

Экспериментальным путем подобрана архитектура нейронной сети, при которой удается прогнозировать с желаемой точностью (0,0001). Полученная архитектура представлена на рисунке 2. Сеть состоит из 3 слоев. Количество входов – 53 (обусловлено входными факторами которых 53); количество нейронов в среднем слое – 6, функции активации - гиперболический тангенс; на выходном слое один нейрон, функции активации – линейная.

В результате некоторых экспериментов получили следующие результаты: за 500 итераций генетического алгоритма нейронная сеть обучилась с ошибкой = 0,000114. При этом из 53 факторов выделено 26. В качестве метода селекции использован турнир (количества особей в туре = 3); метод редукции – элитарная стратегия; вероятность операции мутации = 0,02 и вероятность операции скрещивания = 0,98 (одноточечный кроссинговер); число особей в популяции = 25. Влияние количества факторов на функцию приспособленности = 0,35 (35%), а влияние ошибки обучения = 0,65 (65%).



Среди влияющих факторов информативными оказались следующие:

- информация о матери: образование; количество м2 на человека; возраст на момент первых месячных; длительность и болезненность месячных; возраст на момент первой беременности; роды по счету; плодность текущей беременности; способы контрацепции; перенесенные заболевания; курение; рост и вес; анемия; группа крови и резус фактор; рвота; гестоз;

Рисунок 2. Архитектура нейронной сети.

- информация об отце: возраст; алкоголь;
- информация о ребенке: кормили грудью, искусственное питание или смешанное; количество баллов по шкале Апгар; срок гестации; вес; рост;

Для понижения размерности необходимо построить и обучить автоассоциативную сеть. В результате проведенных экспериментов, оказалось, что наилучшие результаты достигаются при

следующих параметрах сети: количество нейронов во входном и выходном слоях – 53 (обусловлено входными факторами которых 53); количество нейронов в среднем слое – 25 (пониженная размерность входных факторов); количество нейронов во втором и четвертом слое – 60. Функции активации для 5,4 и 2 слоев – гиперболический тангенс, для 3 слоя – гладкая ступенчатая. Архитектура сети представлена на рисунке 3.

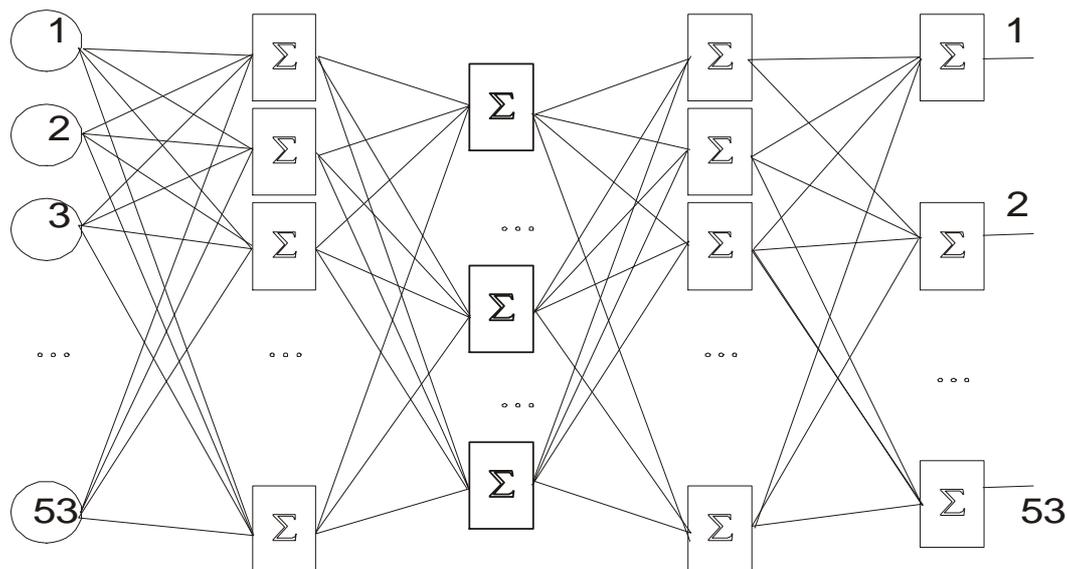


Рисунок 3. Архитектура нейронной сети для понижения размерности: количество нейронов во входном и выходном слоях – 53; количество нейронов в среднем слое – 25; количество нейронов во втором и четвертом слое – 60.

Выводы. При тестировании на реальных медицинских данных после этапа подготовки данных (анализа параметров, кодирования и нормировки) получили обучающий массив размером 53 x 240, где 53 входных параметров и 240 обучающих примеров. Данная выборка разделена на две - по 120 примеров в каждой. По первой выборке проводилось обучение нейронной сети, выбор признаков с помощью генетических алгоритмов и понижение размерности с помощью автоассоциативной сети. В результате проведенных экспериментов количество входных параметров используя генетические алгоритмы удалось сократить до 26, а автоассоциативную сеть - до 25. Затем, на второй выборке проводилось тестирование нейронной сети, в первом случае, с использованием только выделенных, с помощью генетических алгоритмов, факторов, во втором с использованием данных, полученных после сжатия автоассоциативной сетью. Таким образом, результат можно считать положительным. Количество факторов получается уменьшать практически в два раза. Выбор метода понижения размерности зависит от типа экспертной системы для которой готовятся данные. Если экспертная система предназначена реализовывать цепочки рассуждений имитирующих анализ ситуации экспертом человеком, то необходимо использовать метод выделения информативных признаков, а если основную ценность экспертной системы будет представлять собой компьютерная система по методу черного ящика, то можно рассматривать подход, связанный с понижением размерности.

ЛИТЕРАТУРА:

1. Таран Т. А., Зубов Д.А. Искусственный интеллект. Теория и приложения. Учебное пособие. – Луганск: Изд-во ВНУ им. В. Даля, 2006. – 240с
2. Нейронные сети. STATISTICA Neural Networks: Пер. с англ. – М. : Горячая линия – Телеком. 2001. 182 с.
3. Миркес Е.М. Нейроинформатика: Учеб. пособие для студентов / Е.М. Миркес. Красноярск: ИПЦ КГТУ, 2002, 347 с. Рис. 58, табл. 59, библиогр. 379 наименований. Электронный ресурс <http://softcraft.ru/neuro/ni/p07.shtml>
4. Николаев А.Б. Интеллектуальный анализ и обработка данных. Методическое пособие. Электронный ресурс. // <http://web-server.madi.ru/study/kafedra/asu/metod/intellect.html>
5. Саймон Хайкин Нейронные сети: полный курс, 2-е издание. : Пер. с англ. – М. : Издательский дом «Вильямс», 2006. – 1104 с.
6. Рутковская Д., Пилинский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы: Пер. с польск. И.Д. Рудинского. - М.: Горячая линия – Телеком, 2006. – 452 с. : ил.

Ю.А. Скобцов, Т.А. Васяева

Подготовка данных при разработке медицинских экспертных систем

Рассмотрены этапы разработки медицинских экспертных систем и методы подготовки данных для них. Рассмотрены и реализованы методы определения информативной информации из состава факторов риска на основе нейронных сетей и генетических алгоритмов. Разработана архитектура нейронной сети, подобраны методы генетических операторов, разработана фитнес-функция. Проведены исследования и приведены результаты использования методов на реальных медицинских данных.

Ю.О. Скобцов, Т.О. Васяева

Підготовка даних при розробці медичних експертних систем

Розглянуто етапи розробки медичних експертних систем та методи підготовки даних для них. Розглянуто і реалізовано методи визначення інформативної інформації із складу факторів ризику на основі нейронних мереж та генетичних алгоритмів. Розроблена архітектура нейронної мережі, підібрані методи генетичних операторів, розроблена фітнес-функція. Проведені випробування і наведені результати використання методів на реальних медичних даних.

Y.A.Skobtsov, T.A. Vasyaeva

Prepare data for building the medical expert system

Considered development stages of the medical expert systems and methods of data preparation for them. Considered and executed methods of the determination base component from sampling pattern by neural networks and genetic algorithm. Designed architecture neural network, selected methods genetic operators, designed fitness-function. Made experiment and published results of the use the methods on real medical data.