Тема 14

Нейро-нечеткие системы

Применение на практике систем основанных на нечеткой логике оказалось особенно эффективным там, где решающие устройства работают с неточными или неполными входными данными. Однако проблемой таких систем является сложность настройки параметров, обеспечивающих желаемый результат. С другой стороны, очевидны достоинства ИНС в решении трудно формализуемых задач: возможность обучения на примерах и самообучения, толерантность к ошибкам и неточностям входных данных, гибкость структуры, – всё это делает ИНС эффективным инструментом для решения задач близких к задачам, решаемым системами с нечеткой логикой. Соединение этих двух парадигм в одну дает значительные преимущества перед системами, решающими похожие задачи.

Гибридные системы, функционирование которых происходит по правилам нечеткой логики, но реализовано с использованием методов нейроинформатики называются нейро-нечеткими системами (ННС).

Нейронные сети используются для реализации функций фаззификации, функций принадлежности, функций *t*-нормы и *t*-конормы, нечеткой импликации, а также для приведения к четкости результата логического вывода в нечеткой логике. В то же время в рамках нечеткой логики можно при помощи лингвистических переменных закодировать понятия и правила, накопленные в базе знаний экспертов. Возможность обучения таких гибридных систем по любому из алгоритмов обучения ИНС сильно упрощает процесс их разработки и сокращает затраченное время.

Обучение нейро-нечетких систем

Рассмотрим систему правил

$П_1$: если x_1 есть A_{11} И x_2 есть A_{12} ... x_n есть A_{1n}, то y_1 есть B_{11} И y_2 есть B_{12} ... y_k есть B_{1k}
также
$П_2$: если x_1 есть A_{21} И x_2 есть A_{22} ... x_n есть A_{2n}, то y_1 есть B_{21} И y_2 есть B_{22} ... y_k есть B_{2k}
также
...
$П_m$: если x_1 есть A_{m1} И x_2 есть A_{m2} ... x_n есть A_{mn}, то y_1 есть B_{m1} И y_2 есть B_{m2} ... y_k есть B_{mk}

Каждое правило может быть интерпретировано как элемент обучающей выборки для некоторой ИНС: $\{(A_{i1}, A_{i2},\ldots,A_{in}),(B_{i1}, B_{i2}, \ldots, B_{ik})\}\ i=1,2,\ldots,m.$

Метод Умана-Эзава

Существует два подхода к использованию градиентных методов обучения ННС. В первом варианте, предложенном Умана и Эзава, нечеткие множества представляются конечным числом своих представителей.

Пусть [a1,a2] содержит основания всех Aiq, а также основание множества входных векторов системы. Пусть также [b1,b2] содержит основания всех Bip, а также основание множества всех выходных сигналов системы.

Положим что $M>1$, $N>1$ – некоторые целые числа. Пусть

$$x_q=a_1+(q-1)(a_2-a_1)/(N-1),$$
$$y_p=b_1+(p-1)(b_2-b_1)/(M-1).$$
$$1<=p<=M,\ 1<=q<=N.$$

Дискретный вариант обучающей выборки будет выглядеть так:

$$\{(A_{i1}(x_q), A_{i2}(x_q),\dots,A_{in}(x_q)),(B_{i1}(y_p), B_{i2}(y_p), \dots, B_{ik}(y_p)).$$

Для простоты рассмотрим систему следующего вида:

$П_1$: если x есть A_1 , то y есть B_1
также
$П_2$: если x есть A_2 , то y есть B_2
также
...
$П_n$: если x есть A_n , то y есть B_n,

Для которой обучающая выборка имеет вид:

$\{(A_i(x_1), A_i(x_2),\dots, A_i(x_N), B_i(y_1), B_i(y_2), \dots, B_i(y_M))\}$, $i=1,2,\dots,n$.

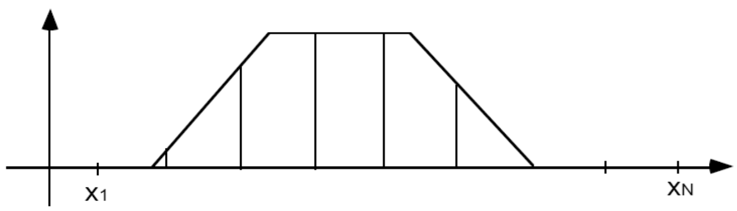

Рис. 2. Представление нечеткого числа набором своих значений

Используя обозначения

$$a_{ij}=A_i(x_j),\ b_{ij}=B_i(y_j),$$

мы приходим к задаче обучения ИНС с N входами и M выходами которую можно обучить одним из градиентных методов.

Второй подход

При втором подходе, для представления нечетких чисел используются конечное число α-уровней.

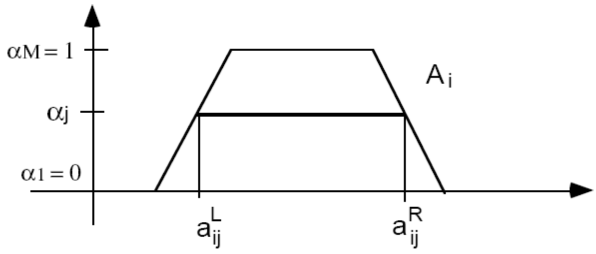

Рис.3. Представление нечеткого числа множеством

Определение 14.1. *Нейро-нечеткой системой* (*сетью*) будем называть ИНС, с четкими входными и выходными сигналами из интервала [0,1] и четкими функциями преобразования сигнала, в которой присутствуют нейроподобные элементы, вычисляющие значения операций нечеткой логики (такие как t-норма, t-конорма и другие).

Основным элементом в такой сети будет *нечеткий нейрон*. Пусть T – треугольная норма, а S – треугольная конорма. Дадим определения основных типов нечетких нейронов.

Определение 14.2. *И нечеткий нейрон* (*AND fuzzy neuron*)– нейрон вычисляющий выходное значение *y* по входному вектору *x* по следующей формуле:

$$y=T(p_1,p_2)=T(S(w_1,x_1),S(w_2,x_2)).$$

К примеру, если T=min, S=max, то y=min(max(w_1,x_1), max(w_2,x_2)).

Определение 14.3. *ИЛИ нечеткий нейрон* (*OR fuzzy neuron*)– нейрон, вычисляющий выходное значение *y* по входному вектору *x* по следующей формуле:

$$y=S(p_1,p_2)=S(T(w_1,x_1),T(w_2,x_2)).$$

К примеру, если T=min, S=max, то y=max(min (w_1,x_1),min (w_2,x_2)).

Определение 14.4. *НЕ нечеткий нейрон* (*NOT fuzzy neuron*)– нейрон, вычисляющий выходное значение *y* по входному значению *x* по следующей формуле:

$$y=1-x.$$

Аналогичным образом можно определить другие операции над нечеткими числами.

Таким образом, из нечетких нейронов мы всегда можем построить ННС моделирующую некоторую систему правил нечеткого вывода {$П_1$, $П_2$, ..., $П_n$}. Результатом работы такой ННС будет значение логического вывода соответствующего этой системе правил.

Пример 14.1. Рассмотрим случай нечеткого вывода по схеме Тсукамото.

Для простоты предположим что мы имеем два «Если … То …» правила следующего вида:

П$_1$: если x есть A_1 и y есть B_1 тогда z есть C_1
также
П$_2$: если x есть A_2 и y есть B_2 тогда z есть C_2
Предпосылка: x есть x_0 и y есть y_0
Вывод: z есть C

Напомним, что формулы для расчета результата вывода всей системы выглядят так:

$$a_1 = [A_1(x_0) \wedge B_1(y_0)],$$

$$a_2 = [A_2(x_0) \wedge B_2(y_0)],$$

$$z_0 = (a_1 z_1 + a_2 z_2)/(a_1 + a_2).$$

При этом настраиваемыми параметрами будут значения z_1 и z_2. Кроме того, мы можем не иметь ясного представления о функции принадлежности нечетких множеств A_1, A_2, B_1, B_2. В этом случае настраиваемыми параметрами могут стать параметры функции принадлежности.

Пример 14.2. Пусть функция принадлежности определяется формулой

$$A(t) = \exp(-\beta(t-1)^2),$$

тогда параметр β будет настраиваемым параметром формулы.

Логический вывод можно реализовать при помощи нейро-нечеткой сети, представленной на рисунке 14.1.

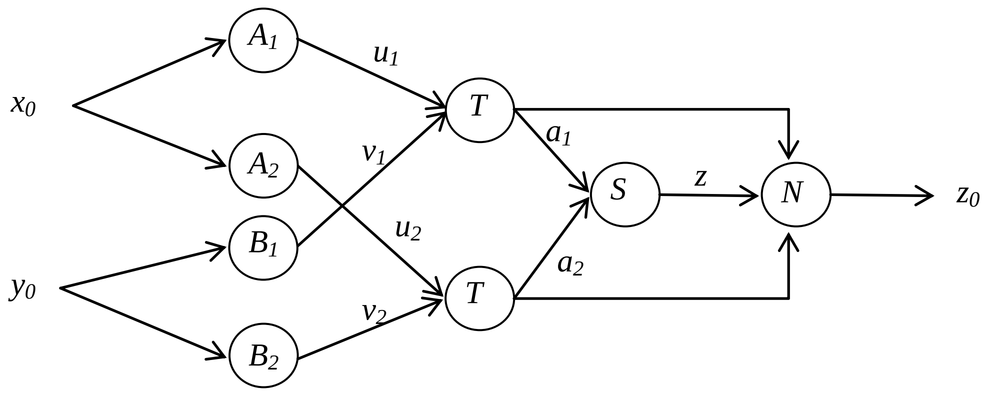

Рис. 14.1. Нейро-нечеткая сеть реализующая вывод по схеме Тсукамото.

В этой сети 4 слоя нейронов и четыре типа нейронов соответственно.

Первый слой – нечеткие нейроны, определяющие степень принадлежности входного сигнала некоторому множеству. Они по входному значению вычисляют характеристическую функцию соответствующего нечеткого множества.

Второй слой – слой нечетких И-нейронов, вычисляющих t-норму от входных сигналов:

$$T:\ a=T(u,v).$$

Третий слой – нейроны, вычисляющие взвешенную сумму входных сигналов:

$$S:\ z=a_1z_1+a_2z_2.$$

Четвертый слой – нейрон осуществляющий приведение к четкости посредством нормирования входного значения:

$$N:\ z_0=z/(a_1+a_2).$$

Таким образом, прохождение сигналов через сеть соответствует последовательности операций совершаемых при нечетком выводе по схеме Тсукамото.

С другой стороны, может возникнуть необходимость подобрать параметры формул вывода. Это касается как параметров нечетких множеств, так и других параметров формул. Подбор параметров может осуществляться методами нейроинформатики. Так, используя метод обратного распространения ошибки, можно настраивать параметры сети одним из градиентных методов обучения.

Пример 14.3. Теперь рассмотрим сеть для нечеткого вывода по схеме Мамдани.

Формула, по которой вычисляется результат вывода, выглядит следующим образом:

$$C(w) = C'_1(w) \vee C'_2(w) = ([A_1(x_0) \wedge B_1(y_0)] \wedge C_1(w)) \vee ([A_2(x_0) \wedge B_2(y_0)] \wedge C_2(w)).$$

Нейро-нечеткая сеть, реализующая вычисления по этой формуле показана на рисунке 14.2.

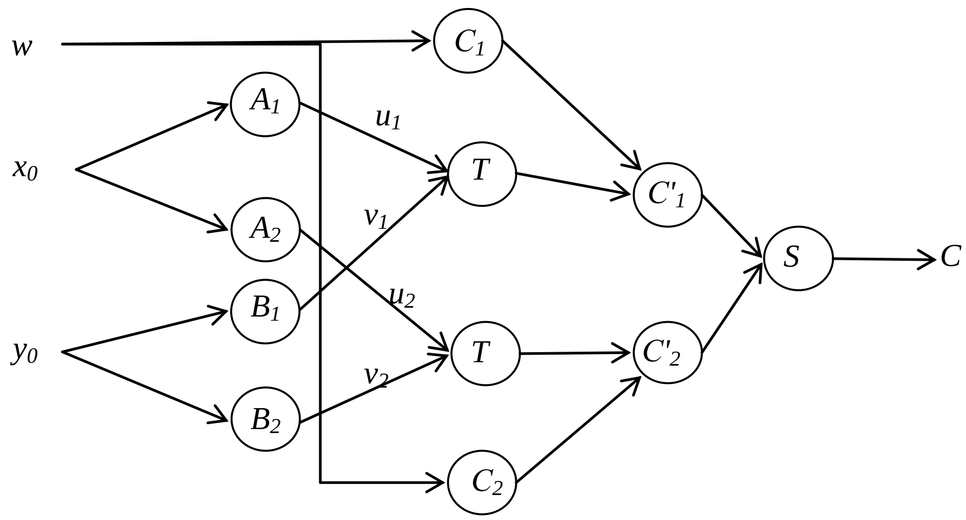

Рис. 14.2. Нейро-нечеткая сеть реализующая вывод по схеме Мамдани.

В этой сети четыре слоя, однако, в отличие от ННС Тсукамото, отсутствует слой дефаззификации. Первый и второй слои аналогичны первому и второму слою ННС Тсукамото. Третий слой вычисляет t-норму от промежуточных значений. Четвертый слой вычисляет t-конорму от значений, полученных на предыдущем слое. Результатом работы сети будет нечеткая переменная $C(w)$. К нему можно применить любой из существующих способов приведения к четкости.

Задание. Построить ННС для схемы вывода Сугено-Такаги.

Литература

1. Robert Fullér, Neural Fuzzy Systems, Abo Akademis tryckeri, Åbo, 1995, 249 pages. ISSN 0358-5654, ISBN 951-650-624-0.
2. Zadeh L.A. Fuzzy sets. - Information and Control, 1965, vol.8, N 3,pp.338-353.
3. Батыршин И.З. Основные операции нечеткой логики и их обобщения. – Казань: Отечество, 2001. - 102 с. (ISBN 5-9222-0034-8).
4. Рутковская Д., Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы. – М., 2004.
5. Круглов В. В., Дли М. И., Голунов Р. Ю. Нечеткая логика и искусственные нейронные сети: Учеб. пособие. -- М.: Издательство Физико-математической литературы, 2001.