3. ЛЕКЦИЯ. Нечеткие алгоритмы обучения

Известно, что обучающиеся системы улучшают функционирование в процессе работы, модифицируя свою структуру или значение параметров. Предложено большое число способов описания и построения обучающихся систем. Все они предполагают решение следующих задач: выбор измерений (свойств, рецепторов); поиск отображения пространства рецепторов в пространство признаков, которые осуществляют вырожденное отображение объектов; поиск критерия отбора признаков. Причем в различных задачах для получения хороших признаков могут понадобиться разные критерии отбора. При обучении необходимо отвлечься от различий внутри класса, сосредоточить внимание на отличии одного класса от другого и на сходстве внутри классов. Необходим достаточный уровень начальной организации обучающейся системы. Для сложной структурной информации необходима многоуровневая обучающаяся система.

Следует выделить следующие группы нечетких алгоритмов обучения:

1.обучающийся нечеткий автомат;

2.обучение на основе условной нечеткой меры;

3.адаптивный нечеткий логический регулятор;

4.обучение при лингвистическом описании предпочтения.

Рекуррентные соотношения в алгоритмах первых двух групп позволяют получать функцию принадлежности исследуемого понятия на множестве заранее известных элементов. В третьей группе нечеткий алгоритм обучения осуществляет модификацию нечетких логических правил для удержания управляемого процесса в допустимых границах. В четвертой группе нечеткий алгоритм обучения осуществляет поиск вырожденного отображения пространства свойств в пространство полезных признаков и модификацию на их основе описания предпочтения.

Обучающийся нечеткий автомат

Рассмотрим автомат с четким входом i(t) и зависимым от времени нечетким отношением перехода δ(t). Пусть ∽s(t) — нечеткое состояние автомата в момент времени t на конечном множестве состояний S=$\{S_1 ,\ldots, S_n\}$, и i_l - оценка значения i(t).Состояние автомата в момент времени (t+1) определяется min - max композицией:

$$\mu_{\sim s(t+1)}(S_k) = \sup\min(\mu_{\sim s(t)}\left(S_j\right), \mu_{\delta(t)}(s_x, i_l, s_j)), \qquad (1)$$

Или аналогично ей. Обучение направлено на изменение нечеткой матрицы переходов:

$$\mu_{\delta(t)}\left(s_k, i_l, s_j\right) = \mu_{\delta(t-1)}\left(s_k, i_l, s_k\right), \text{где } j \neq k$$

$$\mu_{\delta(t)}(s_k, i_l, s_k) = \alpha_{\kappa\mu\delta(t-1)}(s_k, i_l, s_k) + (1 - \alpha_k)\lambda_\kappa(t),$$

где $0 < \alpha_\kappa < 1, 0 < \lambda_\kappa(\mathrm{t}) \leq 1, k = 1, \dots, n$

Константа λ_kопределяет скорость обучения. Начало работы автомата возможно без априорной информации $\mu_{\sim s(0)}(s_k) = 0$ или 1, а также с априорной информацией $\mu_{\sim s(0)}(s_k) = \lambda_k(0)$. Величина $\lambda_k(t)$ зависит от оценки функционирования автомата. Доказано, что имеет место сходимость матрицы переходов, независимо от того, есть ли априорная информация, т.е. $\mu_{\sim s(0)}\left(s_j\right)$ может быть любым значением из интервала [0,1].

Пример. На рис. 1 изображена модель классификации образов.

Рис.1. Модель классификации.

Роль входа и выхода можно кратко объяснить следующим образом. Во время каждого интервала времени классификатор образов получает новый образец x' из неизвестной внешней среды. Далее x' обрабатывается в рецепторе, из которого поступает как в блок "обучаемый", так и в блок "учитель" для оценки. Критерий оценки должен быть выбран так, чтобы его минимизация или максимизация отражала свойства классификации (классов образов). Поэтому, благодаря естественному распределению образов, критерий может быть включен в систему, чтобы служить в качестве учителя для классификатора.

Модель обучения формируется следующим образом. Предполагается, что классификатор имеет в распоряжении множество дискриминантных функций нескольких переменных. Система адаптируется к лучшему решению. Лучшее решение выделяет множество дискриминантных функций, которые дают минимум не распознавания среди множества дискриминантных функций для данного множества образцов.

Алгоритм поиска глобального экстремума приведен на рис.2.

Рис.2. Поиск глобального экстремума.

Моделируется поиск глобального экстремума функции следующим образом:

1. Область определения целевой функции делится на некоторое число подобластей (форма подобластей постоянно меняется) и описывается некоторым множеством точек;
2. каждой точке приписывается состояние автомата, причем функция принадлежности в каждом состоянии указывает степень близости к оптимуму;
3. выбирается состояние с максимальным значением функции принадлежности (эта точка называется кандидатом);
4. формируется новая подобласть из точек, окружающих кандидата (размер подобласти растет, когда значение целевой функции в точке кандидата меньше, чем в других точках подобласти, и уменьшается в противоположном случае);

5. когда подобласть пересекается с некоторой другой, или две точки-кандидаты находятся в одной подобласти, то подобласти разделяются, если степень разделения большая, или объединяются, если степень разделения малая;

6. точки-кандидаты выбираются на этапе локального поиска в подобласти, затем во всей области среди точек-кандидатов ищется глобальная оптимальная точка;

7. глобальный и локальный поиск осуществляется поочередно.

Пример:

Пусть S — множество состояний, V — выходной универсум, δ - функция выхода (функция принадлежности, указывающая степень оптимума в состоянии s), I(t) - текущее значение целевой функции, I_0 - среднее значение I(t).

Используется следующий алгоритм изменения функций перехода и выхода в случае глобального поиска:

если I(t) > I_0, то попытка успешна и $\mu_{\delta(t)}(s_k, s_j) = \alpha_{k\mu\delta(t)}(s_k, s_j) + (1-\alpha)$,

если I(t) ≤ I_0то попытка неудачна и $\mu_{\delta(t+1)}(u_i, s_j) = \alpha\mu_{\delta(t)}(u_i, s_j)$,

где $\alpha = 1 - \left|\frac{(I(t) - I_0)}{I_0}\right|$ $\alpha < 1$- гарантируемая сходимость.

В случае локального поиска:

если I(t)>I_0, то $\mu_{\delta(t+1)}(u_i, s_j) = \alpha\mu_{(t)}(u_i, s_j) + (1-\alpha)$,

если I(t)≤ I_0, то $\mu_{\delta(t+1)}(u_i, s_j) = \alpha\mu_{\delta(t)}(u_i, s_j)$.

Обучение на основе условной нечеткой меры

Пусть X={$x_1, \ldots x_n$}- множество причин (входов) и Y={$y_1, \ldots y_n$} - множество результатов. Если h — функция из X в интервал [0,1], (h(x_1) $\leq \ldots \leq$ $h\ (x_n)$ и g_x} - нечеткая мера на X, то

$\int_x h(x) g_x(\cdot) = \max\limits_{i=1,\ldots,n} \min(h(x_i), \mathrm{gx}(H_i))$, где $H_i = \{x_i, \ldots, x_n\}$.

Задача состоит в оценке (уточнении) причин по нечеткой информации.

Пусть gY — нечеткая мера на Y, gY связана с gX условной нечеткой мерой $\sigma\gamma(\cdot\,|x)$:

$$gY = \int_X \sigma Y(\cdot\,|x) gx$$

Предполагается следующая интерпретация вводимых мер: gX оценивает степень нечеткости утверждения "один из элементов X был причиной", $\sigma Y(A|x)$, A⊂Y оценивает степень нечеткости утверждения «один из элементов A является результатом благодаря причине x»; gY({y}) характеризует степень нечеткости утверждения: «y — действительный результат».

Пусть μ_A(y) описывает точность информации A, тогда по определению

$$gY(A) = \int_X \mu_{\mathrm{A}}\ (y) gx$$

Метод обучения должен соответствовать обязательному условию: при получении информации А нечеткая мера gX меняется таким образом, чтобы $gY(A)$ возрастала. Предположим, что $gx(\cdot)$ и $\sigma Y(\cdot\ |x)$ удовлетворяют -правилу. Пусть $\sigma Y(\cdot\ |x_i)$является убывающей, тогда

$gY(A) = \bigvee_{i=1}^{n}[\sigma Y(A|x_i) \land gx(F_i)]$,

где F_i={$x_1 \dots, x_i$}. При этих условиях существует l:

$gY(A) = \ \sigma Y(A|x_l) \land gx(F_l)$,

$\sigma Y(A|x_l) \land gx(F_l) \geq \sigma Y(A|x_{l-1}) \land gx(F_{l-1})$,

$\sigma Y(A|x_l) \land gx(F_l) \geq \sigma Y(A|x_{l-1}) \land gx(F_{l-1})$.

Обучение может быть осуществлено увеличением тех значений g_i(i=1, … ,n) нечеткой меры gx, которые увеличивают $gY(A)$, и уменьшением тех значений g_i (i=1, … ,n) меры gx, которые не увеличивают $gY(A)$. Можно показать, что на величину $gY(A)$ влияют только такие g_i, что 1 ≤ i ≤ l. Следовательно, нечеткий алгоритм обучения следующий:

$g^i = \alpha g^i + (1-\alpha)\sigma Y(A|x_l)$; I = 1,…,l;

$g^i = \alpha g^i$; i = l+1,…,n/

Параметр α ∈[0,1] регулирует скорость обучения, т.е. скорость сходимости g^i. Чем меньше α, тем сильнее изменяется g^i. В приведенном алгоритме нет необходимости увеличивать g^i больше, чем на $\sigma Y(A|x_l)$, так как большое увеличение g^i не влияет на $gY(A)$. Приведем некоторые свойства модели обучения.Место для формулы.

Свойство 1. Если повторно поступает одна и та же информация, то происходит следующее:

1.1. Новое g^i больше старого g^i (i=1, … ,l) и новое g^i меньше старого g^i (i=l+1, . . . ,n), следовательно, новая мера $gY(A)$ не меньше старой меры $gY(A)$, и новая мера

$gY(A)$= $\sigma\gamma$ (A|x_k) ∧ gx (F_k), k≤ l;

1.2. При предположении $\sigma\gamma$ (A|x_1) > $\sigma\gamma$ (A|x_2), k< l, g^1 сходится к $\sigma\gamma$ (A|x_1) и g^i сходится к 0 для i=2, . . . ,n.

Свойство 2. Если поступает одна и та же информация повторно: $h_A(y) = c$ для всех y, то $\sigma\gamma$ (A|x) $= \int_X\ c\sigma\gamma(\cdot\ |x) = c, \sigma\gamma$ (A)=c∧ gx(X).

Следовательно, l=n и g^i сходится к c для всех i.

Свойство 3. Предельное значение g^i не зависит от начального значения тогда, когда на вход повторно поступает одна и та же информация.

Пример. Рассмотрим модель глобального поиска экстремума неизвестной функции с несколькими локальными экстремумами. Для поиска глобального экстремума формируются критерии в виде некоторых функций:

x_1 — оценивает число точек, проанализированных на предыдущих шагах;

x_2 — оценивает среднее значение функции по результатам предыдущих шагов;

x_3 — оценивает число точек, значение функции в которых принадлежит десятке лучших в своей области;

x_4 — оценивает максимум по прошлым попыткам;

x_5 — оценивает градиент функции.

В описанном случае gx показывает степень важности подмножеств критериев и $\sigma\gamma\ (\{y_j\}|\ x_i)$ оценивает предположение о нахождении экстремума в блоке y_j в соответствии с критерием x_i. Например, $\sigma\gamma\ (\{y_j\}|\ x_i)$ может зависеть от числа ранее проанализированных точек в блоке y_j. Пусть входная информация A определяется формулой:

$$\mu_A(y_j) = p_j - \frac{\min\limits_k p_k}{\max\limits_k p_k - \min\limits_k p_k},$$

где p_k — максимум анализируемой функции, найденный к рассматриваемому моменту в блоке y_j. Очевидно, что A сходится к максимизирующему множеству функции. На каждой итерации осуществляется следующее: проверяется заданное число новых точек; число этих точек выбирается пропорционально $gY(\{y_j\})$; в каждой точке y_jвычисляется и нормализуется мера $\sigma\gamma(\cdot|x_i)$, нормализуется gx; по $\sigma\gamma$ и σx вычисляется gy($\{y_j\}$), а затем gy(A); посредством правил подкрепления корректируется gy($\{x_i\}$). Затем выполняется новая итерация, и так до тех пор, пока не сойдется gy.

Адаптивный нечеткий логический регулятор

В настоящее время наиболее широкое применение при решении практических задач получили нечеткие логические регуляторы, которые позволяют на основании лингвистической информации, полученной от опытного оператора, управлять сложными, плохо формализованными процессами.

Структура нечеткого логического регулятора, в котором используются эвристические правила принятия решений, показана на рис. 3 Такие регуляторы применяются аналогично традиционным регуляторам с обратной связью. Определение управляющих воздействий состоит из четырех основных этапов:

Рис. 3. Структура нечеткого логического регулятора

Получение отклика;

Преобразование значения отклонения к нечеткому виду, такому, как "большой", "средний";

Оценка входного значения по заранее сформулированным правилам принятия решения с помощью композиционного правила вывода;

Вычисление детерминированного выхода, необходимого для регулирования процесса.

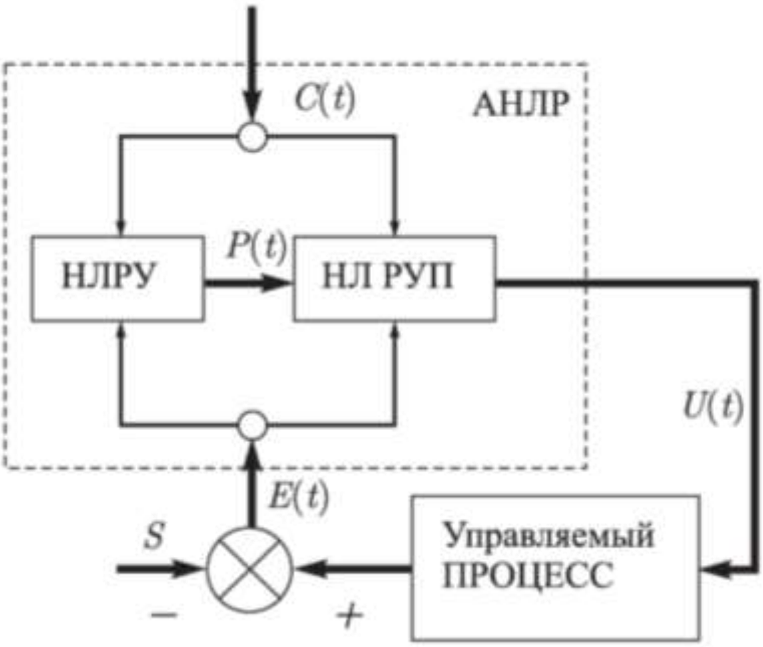

Рис. 4. Адаптивный нечеткий логический регулятор

Опишем способ уточнения правил управления, используемых в адаптивном нечетком логическом регуляторе (АНЛР). Соответствующая схема регулятора приведена на рис. 4 АНЛР состоит из двух частей: нечеткого логического регулятора управляемого процесса (НЛРУП) и нечеткого логического регулятора управления (НЛРУ). На рис. 4 используются следующие обозначения:

U(t) – управление, генерируемое НЛРУП;

E(t) – ошибка отклонение от устанавливаемого выходного значения процесса s ;

S – желаемое значение выхода управляемого процесса, C(t)= E(t) - E(t - 1) ;

P(t) – модификация управления.
Правила НЛРУП имеют форму: if E=E_i then if C=C_i then U=U_i.
Правила НЛРУ имеют форму: if E=E_j then if C=C_j then P=P_j.

Здесь E_i, E_j, C_i, C_j, U_i, U_j — предварительно описанные нечеткие множества. Символ P(t) используется для модификации стратегии управления следующим образом: в нечетком правиле i, которое ухудшает течение процесса, заменяется значение управления U на $U_i' = U_i \otimes P_i(t)$. Правило i в НЛРУП заменяется на правило if E=E_i then if C = C_i then U=U_i'.

Рассмотрим далее два нечетких алгоритма обучения при лингвистическом описании предпочтений: алгоритм формирования нечеткого отношения предпочтений на множестве альтернатив, описываемых наборами лингвистических значений признаков, и алгоритм уточнения лингвистических критериев.

Алгоритм формирования нечеткого отношения предпочтения

Пусть R — множество таких альтернатив, что каждое S ∈ R характеризуется набором оценок по n признакам: S=$\{t_1, \ldots t_n\}$, и пусть B — семейство всех непустых конечных подмножеств множества R. Для некоторого R' ∈ B известно подмножество выбранных альтернатив R'' ⊂ R', т.е. для любых S'' ∈ R'' и S' ∈ R'\R'' имеет место доминирование S'' ≺ S'. Предварительно, при анализе исходного множества альтернатив, сформирован эталонный набор нечетких оценок $A^0 = (t_1^0, \ldots t_n^0)$. Значения функции принадлежности нечеткой оценки t_1^0 указывают на степень близости значений i -го признака к значениям, определяющим идеальную альтернативу. Используя множество предпочтений

E = {(S'',S'): S'' ∈ R'', S' ∈ R'\R''},требуется найти обобщенные правила предпочтения на множестве R.

Пример. Рассмотрим задачу выбора для рыболовецкого судна рационального района промысла с учетом следующих показателей: u_1 − время перехода в район лова, u_2 − прогноз вылова, u_3 − стоимостная характеристика прогнозируемого объекта лова, u_4 − гидрометеоусловия. Показатели, в сущности, играют роль лингвистических переменных.

Лицу, принимающему решение, предложены альтернативы S_1- S_6(табл.1). Пусть выбрана альтернатива S_1. Для обучения формируются две таблицы:

K_1= $\{(S_1, S_2),(S_1, S_3),(S_1, S_4),(S_1, S_5),(S_1, S_6)\}$

K_2 = $\{(S_2, S_1),(S_3, S_1),(S_4, S_1),(S_5, S_1),(S_6, S_1)\}$

Таблица 1.

	U_1	U_2	U_3	U_4

S_1	хорошо	хорошо	хорошо	удовлетворительно
S_2	очень хорошо	плохо	хорошо	удовлетворительно
S_3	очень хорошо	хорошо	хорошо	неудовлетворительно
S_4	удовлетворительно	хорошо	хорошо	удовлетворительно
S_5	очень хорошо	хорошо	хорошо	удовлетворительно
S_6	хорошо	нормально	плохо	удовлетворительно

	U_1	U_2	U_3	U_4
S_1	плохо	хорошо	плохо	удовлетворительно
S_2	удовлетворительно	хорошо	хорошо	неудовлетворительно
S_3	плохо	хорошо	хорошо	удовлетворительно
S_4	удовлетворительно	хорошо	нормально	удовлетворительно
S_5	удовлетворительно	нормально	нормально	удовлетворительно
S_6	-	-	-	-

Для каждой пары наборов $(S_i\,,S_j)$ вычисляются оценки сравнения i -го элемента первого набора с i -м элементом второго набора:

$$\left.\begin{matrix}(t_1',\dots,t_n')\\ \\ (t_1'',\dots,t_n'')\end{matrix}\right\} \rightarrow (L^{\alpha}(t_1',t_n''),\dots L^{\alpha}(t_1',t_n''))$$

где α определяет конкретный оператор, например, нечеткую меру сходства.

В результате получаются две таблицы наборов нечетких оценок поэлементного сравнения. На основе полученных таблиц, используя логические операторы и логические функции двух переменных, выделяются полезные признаки и минимальный базис. Содержательное значение утверждения, соответствующего минимальному базису, следующее:

$$\Phi(S_i\,,S_j) \succ \Phi(S_j\,,S_i) = (x_1^i \succ\ x_1^j)\&(x_2^i \succ\ x_2^j)\ \&(x_4^i \succ\ x_4^j) \succ (x_1^i \prec\ x_1^j)\&(x_2^i \prec\ x_2^j)\ \&(x_4^i \prec\ x_4^j)$$

где x_k^m - лингвистическое значение k -го показателя, Ф - логический признак. Физический смысл приведенного утверждения: район S_i предпочтительнее района S_j, если утверждение [(время перехода до S_i "меньше", чем до S_j), и (прогноз вылова в S_i "больше", чем в S_j), и (погодные условия в S_i"лучше", чем в S_j)] более истинно, чем обратное утверждение [(время перехода до S_i"больше", чем до S_j), и (прогноз вылова в S_i"меньше", чем в S_j), и (погодные условия в S_i"хуже", чем в S_j)].

Далее предположим, что среди неизвестных ситуаций S_7 - S_{11} (табл. 1) необходимо выбрать лучшую альтернативу, используя минимальный базис. В табл. 2 изображена матрица предпочтений M=(μ^{ij} (K_1) | μ^{ij} (K_2)), элементы которой вычислялись посредством гарантированной оценки

$$\mu^{ij}\ (K_1) = \max_{\upsilon\in[0,1]} \mu H_1\,(S_i,S_j)(\upsilon),$$

Таблица 2.

	S_7	S_8	S_9	S_{10}	S_{11}
S_7		0,88 0,38	1 0,38	0,88 0,38	0,88 0,38
S_8	0,75 1		0,75 1	0,75 1	0,75 1
S_9	1 0,38	0,88 0,38		0,88 0,38	0,88 0,38
S_{10}	1 0,38	1 0,38	1 0,38		1 0,38
S_{11}	0,88 0,38	0,88 0,38	0,88 0,38	0,88 0,38	

$H_1\ (S_i,S_j)$ = ⋂($C_j^i \cap C_j\ \ (S_i,S_j\ )$), $C_j\ \ (S_i,S_j)$,

где значение j -го признака на паре альтернатив (S_i,S_j), C_j^i - значение j -го признака на парах альтернатив i -го класса (i=1,2). Каждый элемент матрицы содержит два значения. Левое значение указывает степень, с которой S_i доминирует над S_j. Правое значение указывает степень, с которой S_j доминирует над S_i. Для построения нечеткого графа предпочтений альтернатив (рис.5) используется следующее правило определения отношения доминирования D:

$$\mathrm{D}(S_i,S_j) = \begin{cases} S_i \succ S_j, \text{ если } \mu_1 \geq \mu_2; \\ S_j \succ S_i, \text{ если } \mu_1 \leq \mu_2; \end{cases}$$

где $\mu_1 = \mu^{ij}\ (K_1) \vee \mu^{ji}\,(K_2), \mu_2 = \mu^{ij}\ (K_2) \vee \mu^{ji}\,(K_1)$.

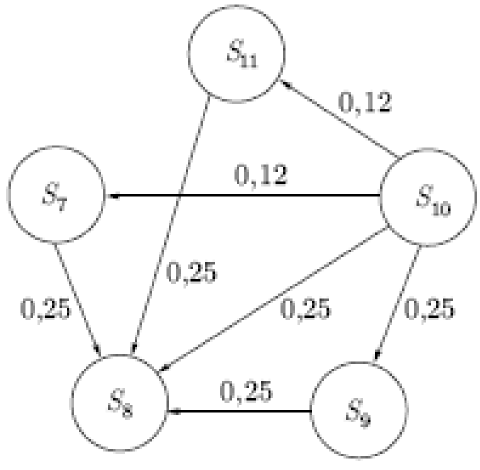

Рис. 5. Нечеткий граф предпочтений альтернатив

Согласно рис. 12.5, S_{10} является недоминируемой альтернативой, т.е. не существует альтернативы, которая с ненулевой степенью доминирует над S_{10}.