

«Оптимизация нечетких нейронных продукционных моделей с помощью кластерного анализа»

Натёкин Алексей Геннадьевич, гр. 522

Санкт-Петербургский государственный университет
Математико-механический факультет
Кафедра статистического моделирования

Научный руководитель: к.ф.-м.н., доцент Пономарева А.Ю.
Рецензент: к.ф.-м.н. Шпилёв П.В.

Санкт-Петербург
2011г.

Цели работы:

- Исследование методов автоматической инициализации всех компонент нечетких нейронных продукционных моделей;
- Рассмотрение нечетких продукционных правил с многомерными функциями принадлежности;
- Разработка обобщения нечетких нейронных продукционных моделей для случая многомерных функций принадлежности;
- Сравнение эффективности предложенных моделей с другими методами машинного обучения на практических задачах.
- Реализация предложенных моделей на языке R.

Постановка задачи:

- $(x, y)_k$ - обучающая выборка, $k = \overline{1, N}$;
- x_k - k -ый вектор входных переменных, $x_k \in \mathbb{R}^m$;
- y_k - k -ое значение выходной переменной, $y_k \in Y$.

Требуется построить функцию f :

$$f : \mathbb{R}^m \rightarrow Y, y_k = f(x_k) \forall k = \overline{1, N}.$$

В зависимости от типа выходной переменной, решаются задачи:

- $Y = \mathbb{R}$ - решается задача регрессии;
- $Y = \{y_1, \dots, y_p\}$ - решается задача классификации.

Нечеткая продукционная модель - набор согласованных правил вида:

ЕСЛИ x есть A , **ТО** y есть B

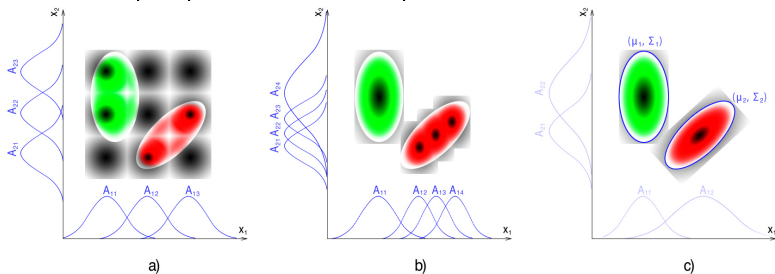
- A - предпосылка правила (антецедент);
- B - заключение правила (консеквент).

Для построения нечеткой продукционной модели определим:

- базу нечетких продукционных правил;
- способ (схему) нечеткого вывода заключений и вид правил;
- процедуры агрегирования, активизации и дефаззификации.

Задание базы нечетких продукционных правил

Инициализация базы нечетких продукционных правил с помощью разбиения пространств входных переменных:



a) Разбиение пространств входных переменных на решетку гиперкубов.

Π_{ij} : ЕСЛИ x_1 есть A_{1i} И x_2 есть A_{2j} ТО ... , $i = \overline{1,3}$, $j = \overline{1,3}$.

b) Разбиение каждого из пространств входных переменных с помощью группировки одномерных функций принадлежности.

Π_i : ЕСЛИ x_1 есть A_{1i} И x_2 есть A_{2i} ТО ... , $i = \overline{1,4}$.

c) Разбиение с помощью многомерных функций принадлежности.

Π_i : ЕСЛИ x есть A_i ТО ... , $x = [x_1, x_2]^T$, $i = \overline{1,2}$.

Модель Такаги-Сугэно:

Нечеткие продукционные правила имеют вид:

Π_i : **ЕСЛИ** x_1 есть A_{i1} **И** ... **И** x_m есть A_{im} **ТО**

$$\hat{y}_i = c_{i0} + \sum_{j=1}^m c_{ij}x_j, i = \overline{1, G}, j = \overline{1, m}, \text{ где}$$

- A_{ij} - нечеткие множества, определенные на X_j с функциями принадлежности $\mu_{A_{ij}}(x_j) \in [0, 1]$, $i = \overline{1, G}$, $j = \overline{1, m}$;
- $\mu_{A_{ij}}(x_j) = e^{-\frac{(x_j - a_{ij})^2}{2b_{ij}^2}}$ - функция принадлежности j -ой входной переменной i -го правила $i = \overline{1, G}$, $j = \overline{1, m}$;

Агрегирование: произведение функций принадлежности.

Активизация: вычисление $\hat{y}_i = c_{i0} + \sum_{j=1}^m c_{ij}x_j, i = \overline{1, G}, j = \overline{1, m}$.

Дефаззификация: среднее взвешенное значение.

$$\hat{y} = \frac{\sum_{i=1}^G \left((c_{i0} + \sum_{j=1}^m c_{ij}x_j) \prod_{j=1}^m \mu_{A_{ij}}(x_j) \right)}{\sum_{i=1}^G \prod_{j=1}^m \mu_{A_{ij}}(x_j)}.$$

Модель Такаги-Сугэно с многомерными функциями принадлежности:

$$\prod_{j=1}^m \mu_{A_{ij}}(x_j) = \prod_{j=1}^m e^{-\frac{(x_j - a_{ij})^2}{2b_{ij}^2}} = e^{-\frac{1}{2}(x - a_i)^T \begin{pmatrix} \frac{1}{b_{i1}^2} & 0 & \dots & 0 \\ 0 & \frac{1}{b_{i2}^2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \frac{1}{b_{im}^2} \end{pmatrix} (x - a_i)} =$$
$$= e^{-\frac{1}{2}(x - a_i)^T \Sigma_i^{-1} (x - a_i)} = \mu_{\hat{A}_i}(x), \text{ где}$$

- $x = [x_1, \dots, x_m]^T$ - вектор входных переменных;
- $\mu_{\hat{A}_i}(x)$ - многомерная функция принадлежности i -го правила, $i = \overline{1, G}$, $\mu_{\hat{A}_i}(x) \in [0, 1]$;
- $a_i = [a_{i1}, \dots, a_{im}]^T$ - вектор средних значений многомерной функции принадлежности i -го правила, $i = \overline{1, G}$;
- Σ_i - ковариационная матрица многомерной функции принадлежности i -го правила, $i = \overline{1, G}$.

⇒ Далее будут рассматриваются произвольные ковариационные матрицы, т.е. без ограничения на некоррелированность входных переменных.

Модель Такаги-Сугэно с многомерными функциями принадлежности:

Функциональная зависимость нечеткой продукционной модели:

$$\hat{y} = \frac{\sum_{i=1}^G \left((c_{i0} + \sum_{j=1}^m c_{ij} x_j) \mu_{\hat{A}_i}(x) \right)}{\sum_{i=1}^G \mu_{\hat{A}_i}(x)}.$$

Будет рассматриваться модель с нормировкой на объемы:

$$\hat{y} = \frac{\sum_{i=1}^G \left((c_{i0} + \sum_{j=1}^m c_{ij} x_j) \frac{\mu_{\hat{A}_i}(x)}{V_i} \right)}{\sum_{i=1}^G \frac{\mu_{\hat{A}_i}(x)}{V_i}},$$
$$V_i = (2\pi)^{\frac{m}{2}} |\Sigma_i|^{\frac{1}{2}}, i = \overline{1, G}.$$

⇒ Для задания базы нечетких правил, требуется найти разбиение входных данных на многомерные нормальные распределения.

Пусть данные порождаются смесью вероятностных распределений:

$$p(x) = \sum_{j=1}^G \phi_j f_j(x|\theta_j), \text{ где}$$

- $x = [x_1, \dots, x_m]^T, x \in \mathbb{R}^m$ - вектор входных переменных;
- $f_j(x|\theta_j)$ - функция плотности распределения j -ой компоненты смеси распределений, $f_j(x|\theta_j) \sim \mathcal{N}(a_j, \Sigma_j)$;
- θ_j - параметры плотности j -ой компоненты смеси, $\theta_j = \{a_j, \Sigma_j\}$;
- ϕ_j - нормализующий вес j -ой компоненты смеси, $\sum_{j=1}^G \phi_j = 1$.

ЕМ алгоритм: вводятся скрытые параметры $w_{kj}, j = \overline{1, G}, k = \overline{1, N}$.

- На Е-шаге по текущим значениям параметров θ_j, ϕ_j вычисляются ожидаемые значения скрытых параметров w_{kj} .
- На М-шаге по функции правдоподобия всех данных максимизируются параметры θ_j, ϕ_j .

$$\theta^{t+1} = \arg \max_{\theta} L(\theta^t, \phi^t, w^t|x)$$

$$L(\theta_j, \phi_j, w_{kj}|x) = \prod_{j=1}^N \log(p(x_j)).$$

Байесовский информационный критерий

Модель с оптимальным числом кластеров выбирается с помощью Байесовского информационного критерия:

$$BIC_k = 2 \log(L(x, \hat{\theta}_k)) - n_{par} \log(N), k \in G_{init}, \text{ где}$$

- $\log(L(x, \hat{\theta}_k))$ - логарифм максимума функции правдоподобия для k -ой модели;
- N - число точек, по которым оценивается модель;
- n_{par} - число независимых параметров k -ой оцениваемой модели;
- G_{init} - предполагаемое число кластеров.

Ищется баланс между приростом правдоподобия и штрафом на добавление параметров для модели.

Выбирается модель, для которой достигается максимум информационного критерия:

$$p(x, \theta)_{result} = p(x, \theta_{k_{result}}, \phi_{k_{result}}), \\ k_{result} = \arg \max_{k \in G_{init}} BIC_k.$$

Проведем разложение ковариационных матриц многомерных функций принадлежности по собственным векторам(EVD):

$$\Sigma_i = U_i \Lambda_i U_i^T = \sum_{j=1}^m \lambda_{ij} u_{ij} u_{ij}^T, \quad U_i = \begin{pmatrix} | & & | \\ u_{i1} & \cdots & u_{im} \\ | & & | \end{pmatrix}$$

Можно представить ковариационную матрицу Σ_i i -го правила, $i = \overline{1, G}$, в виде двух групп параметров:

- матрицы поворота U_i , составленной из собственных векторов u_{ij} ковариационной матрицы Σ_i , $j = \overline{1, m}$;
- вектора, состоящего из собственных чисел λ_{ij} ковариационной матрицы Σ_i , $j = \overline{1, m}$.

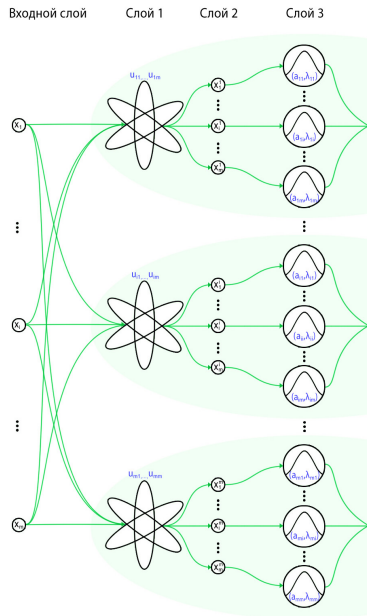
⇒ Вместо параметров ковариационной матрицы Σ_i будем хранить в нейронной сети разложение на собственные вектора u_{ij} и собственные числа λ_{ij} , $j = \overline{1, m}$.

Слой 1: содержит матрицы поворота U_i , полученные из EVD-разложения Σ_i , $i = \overline{1, G}$.
Входной вектор $x = [x_1, \dots, x_m]^T$ переводится в новые координаты:

$$\tilde{x} = U_i x.$$

Слой 2: содержит повернутые значения входных переменных для каждого из правил.

Слой 3: состоит из элементов, вычисляющих функции принадлежности $\mu_{\hat{A}_i}(\tilde{x})$ для данных после поворота в первом слое.
В этом слое хранятся параметры a_i и λ_{ij} функций принадлежности.



Структура нейронной сети

Слой 4: состоит из элементов, вычисляющих значения многомерных функций принадлежности нечетких продукционных правил:

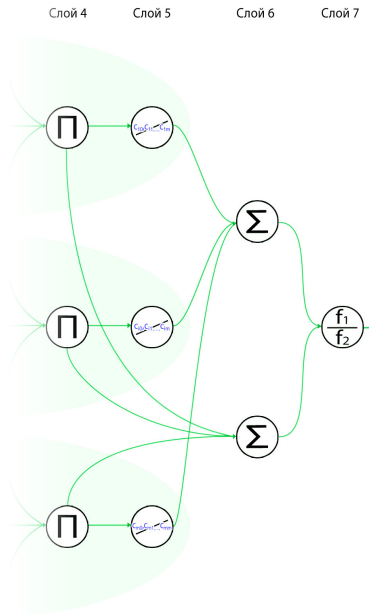
$$\frac{\mu_{\hat{A}_i}(x)}{V_i}, \quad i = \overline{1, G}.$$

Слой 5: элементы данного слоя вычисляют значения консеквентов каждого из правил:

$$\hat{y}_i = \left(c_{i0} + \sum_{j=1}^m c_{ij} x_j \right), \quad i = \overline{1, G}.$$

Слой 6: состоит из двух элементов-сумматоров.

Слой 7: состоит из одного нормализующего элемента.



Гибридный алгоритм обучения нейронной сети

Требуется модифицировать параметры $\theta = \{a, U, \Lambda, c\}$ нечеткой продукционной модели, чтобы минимизировать ошибку:

$$E = \sum_{k=1}^N \frac{1}{2} E^k, \\ E^k = (\hat{y}_k - y_k)^2, \text{ где}$$

\hat{y}_k - значение, полученное нечеткой продукционной моделью.

Гибридный алгоритм обучения:

- С помощью МНК оптимизируем параметры c_{ij} , $i = \overline{1, G}$, $j = \overline{0, m}$.
- С помощью градиентного спуска, оптимизируем параметры λ_{ij} , a_{ij} , $i = \overline{1, G}$, $j = \overline{1, m}$:

$$a_{ij}^k(t+1) := a_{ij}^k(t) - \eta \frac{\partial E^k(t)}{\partial a_{ij}^k}, \\ \lambda_{ij}^k(t+1) := \lambda_{ij}^k(t) - \eta \frac{\partial E^k(t)}{\partial \lambda_{ij}^k}, k = \overline{1, N}, \text{ где}$$

t - номер итерации оптимизационного процесса;

η - константа, регулирующая скорость градиентного спуска.

Геометрический смысл: расщепляются те компоненты нечеткой продукционной модели, на которых ошибка максимальна.

$$E_p = \sum_{k=1}^N \frac{1}{2} E^k \delta_{qp}, \text{ где}$$

- δ_{qp} - символ Кронекера, $p = \overline{1, G}$;
- $q = \arg \max_{q \in \overline{1, G}} \mu_{\hat{A}_q}(x_k)$ - правило с максимальной принадлежностью.

Расщепляется компонента с максимальной ошибкой $\hat{q} = \arg \max_{p \in \overline{1, G}} E_p$.

Выберем $\mu_{\hat{A}_{\hat{q}}}$ и те точки x_k , $k = \overline{1, N}$, для которых выполняется:

$$q = \arg \max_{q \in \overline{1, G}} \mu_{\hat{A}_q}(x_k).$$

Найденные точки komponуются в пары $(x, y)_q$ и для них проводится ЕМ-кластеризация на 2 кластера.

Критерии остановки алгоритма гибридного обучения нейронной сети:

- $E = \sum_{k=1}^N \frac{1}{2} E^k < E_{max}$, где E_{max} - целевая точность, которая вводится исследователем.
- Выполнено правило остановки итерационного процесса:

$$|E^k(t) - E^k(t-1)| \leq \varepsilon, \text{ где}$$

ε - чувствительность к приращению целевой функции.

- Выполнено максимально допустимое число итераций алгоритма обучения: $t > t_{max}$.

Критерии остановки алгоритма разрастания нейронной сети:

- Число компонент $G \geq G_{max}$, где G_{max} - максимальное число правил в нейронной сети.
- Прирост точности при увеличении числа компонент недостаточен:

$$|\check{E}(g) - \check{E}(g-1)| \leq \varepsilon, g = \overline{G_{+1}, G_{max}}, \text{ где}$$

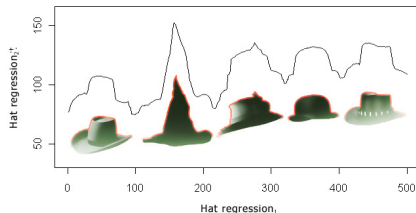
ε - минимальное допустимое увеличение точности.

$\check{E}(g)$ - значение целевой функции ошибки нейронной сети с g компонентами, после алгоритма гибридной оптимизации.

Рассмотрим функцию, полученную из силуэтов шляп с разных фотографий.

$$y = f(\mathbf{x} \mid \theta) : \mathbb{R} \rightarrow \mathbb{R},$$

$$\theta = \{\theta_i\}_i = \{a_i, U_i, \Lambda_i, c_i\}_i, i = \overline{1, G}.$$



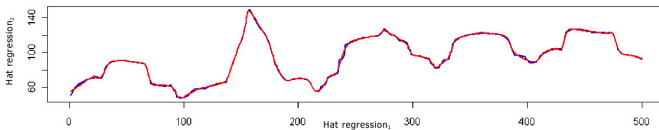
$$\hat{y} = \sum_{i=1}^G \frac{\mu_{A_i}}{\sum_{i=1}^G \mu_{A_i}} \hat{y}_i = \sum_{i=1}^G f_i(x, a_i, U_i, \Lambda_i, c_i) = \sum_{i=1}^G f_i(x, \hat{\theta}_i),$$

$$\hat{\theta} = \arg \min_{\theta} \frac{1}{N} \sum_{k=1}^N (y_k - \hat{y}_k)^2.$$

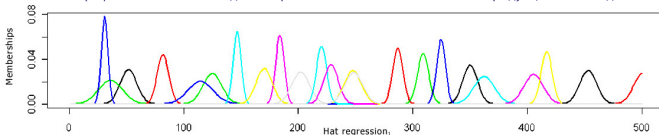
Сравнение эффективности методов машинного обучения

Метод	Число параметров	Ошибка E
Дерево регрессии	25 листьев	19.20510
Нейронная сеть	48 нейронов	4.011456
Машина опорных векторов	90 опорных векторов	3.067771
Предложенная модель, $G = 21$	21 правило	1.499400
Предложенная модель, $G = 24$	24 правила	1.105191

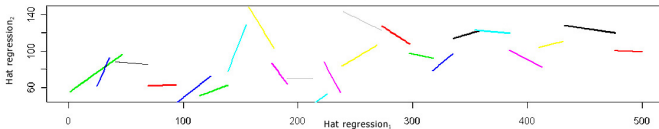
Результаты применения к задаче регрессии



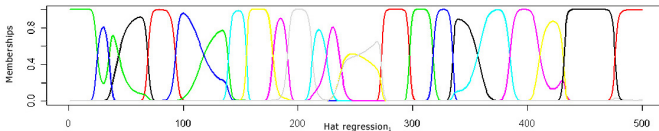
Графики значений выходной переменной и значений нечеткой продукционной модели



Нормированные на объемы нечетких множеств значения функций принадлежности



Локальные линейные функции консеквентов правил

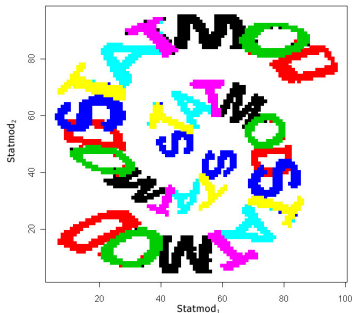


Результирующие значения функций принадлежности, используемые при дефаззификации

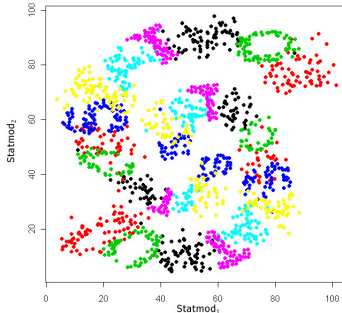
Задача классификации

Рассмотрим качественное усложнение данных двух спиралей:

$$y = f(\mathbf{x} \mid \theta) : \mathbb{R}^2 \rightarrow Y, \\ Y = \{S, T_1, A, T_2, M, O, D\}, \\ \theta = \{\theta_i\}_i = \{a_i, U_i, \Lambda_i\}_i, i = \overline{1, G}.$$



Оригинальные данные



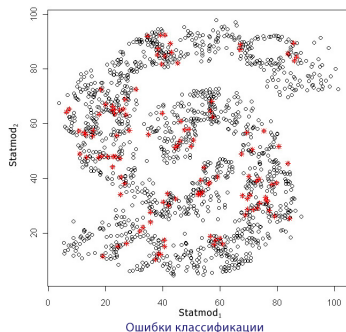
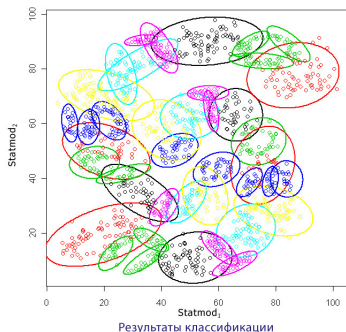
Зашумленные и сокращенные данные

$$\hat{y} = Class_{result}, \quad result = \arg \max_{i \in \overline{1..G}} \mu_{\hat{A}_i}(x, \hat{\theta}_i),$$

$$\hat{\theta} = \arg \min_{\theta} \frac{1}{N} \sum_{k=1}^N \|y_k - \hat{y}_k\|,$$

$$\|y_k - \hat{y}_k\| = \begin{cases} 0 & \hat{y}_k = y_k \\ 1 & \text{иначе} \end{cases}.$$

Результаты применения к задаче классификации



Сравнение эффективности методов машинного обучения

Метод	Число параметров	Ошибка, %
Дерево классификации	54 листа	23.3755
Нейронная сеть	34 нейрона	14.4841
Машина опорных векторов	538 опорных векторов	10.8466
Предложенная модель, $G = 37$	37 правил	9.3254
Предложенная модель, $G = 41$	41 правило	7.5396

Задача проката стали

27-мерные данные проката стали

$T_{кп}$	$T_{см}$	C	Mn	...	Скорость клетки 12	Наличие дефекта
837	653	0.065	0.41	...	683	0
833	656	0.065	0.41	...	720	0
⋮	⋮	⋮	⋮	...	⋮	⋮
843	662	0.068	0.38	...	680	1

- 26 количественных входных переменных задают производственные параметры проката стали;
- 1 бинарная переменная - метка, показывающая, имеются ли на данном листе стали дефекты.

Сравнение эффективности методов машинного обучения

Метод	Число параметров	Ошибка, %
Дерево классификации	152 листа	11.717
Нейронная сеть	38 нейронов	3.8523
Машина опорных векторов	1118 опорных векторов	4.4943
Предложенная модель, $G = 9$	9 правил	2.5682

- Исследован эффективный способ инициализации нечетких нейронных продукционных моделей с помощью ЕМ-алгоритма и Байесовского информационного критерия.
- Построено нейросетевое представление нечетких продукционных моделей с многомерными функциями принадлежности.
- Предложены алгоритмы гибридного обучения нечетких продукционных моделей с многомерными функциями принадлежности.
- Получены успешные результаты применения предложенных моделей как для задачи регрессии, так и для задачи классификации.
- Реализована система функций на языке R.