ИСПОЛЬЗОВАНИЕ НЕЯВНО ЗАДАННЫХ ПЕРЕДАТОЧНЫХ ФУНКЦИЙ (НПФ) ПРИ ПОСТРОЕНИИ ИСКУССТВЕННЫХ НЕЙРОННЫХ СЕТЕЙ

Олександрук Б.О., ННК "Институт прикладного системного анализа" при НТУУ ("Киевский политехнический институт").

В настоящий момент уже исследовано значительное множество моделей нейронных сетей. Ряд моделей нейронных сетей успешно применяются в различных отраслях. Большинство нейросетевых моделей призваны описывать закономерности между входным потоком данных и выходным потоком. Адекватность подбора закономерности нейронной сетью зависит, прежде всего, от структуры сети и передаточных функций нейронов сети. Подробнее остановимся на значимости передаточных функций нейронной сети.

Почему передаточные функции нейронов сети так важны? Дело в том, что закономерность, моделируемая нейронной сетью, представляет собой некоторую суперпозицию передаточных функций нейронов сети. Следовательно, важно, чтобы каждая передаточная функция адекватно описывала на микроуровне поведение элементов реальной системы, сгенерировавшей исходные потоки данных.

Возьмем часто используемые сигмовидные передаточные функции. Нередко возникают ситуации, когда для адекватного описания элементарных закономерностей такими функциями требуется бесконечное число элементов сети. То же самое касается и полиномиальных передаточных функций.

Приведем простой пример.

Пусть искомая закономерность соответствует

$$y = \frac{\sqrt{1-a^2x^2}}{b}$$

т.е. половине эллипса.

Для того чтобы представить данную функцию в виде суперпозиции сигмовидных функций либо полиномов требуется бесконечное число элементов. С другой стороны, данную функцию можно получить как решение из тривиальной модели

$$ax^2+by^2=1.$$

Таким образом, сделаем следующий вывод: используя неявную модель можно добиться большей точности при меньшем количестве степеней свободы (параметров) модели, нежели при использовании явно заданной модели.

Следовательно, замена явно заданной передаточной функции нейронов является не бесперспективной.

Однако с переходом на неявно заданные функции возникает проблема множественных корней, т.е. уравнения модели передаточной функции может иметь несколько корней. Конечно, можно выбирать в качестве решения один из корней, удовлетворяющий определенным условиям. В этом случае функция, которую моделирует нейронная сеть, может иметь разрывы. Чтобы избежать вышеуказанной проблемы рекомендуется использовать уравнение передаточной функции, имеющее единственное решение.

Далее целью работы будет построение уравнения передаточной функции, которое имеет единственное решение.

Лемма 1.

Пусть передаточная функция $y = y(\bar{x}, \bar{a})$ задана уравнением

$g(y, \bar{a}) = f(\bar{x}, \bar{a}) + 1, \qquad (*)$

где $\bar{a}$ – вектор параметров передаточной функции,

y – значения выхода нейрона,

$\bar{x}$ – вектор значений входов нейрона.

$g(y, \bar{a})$, – непрерывная дифференцируемая

$$\frac{\partial g(y, \bar{a})}{\partial y} > 0 \qquad y \in Y$$

Тогда на области $y \in Y$ существует единственное решение

$y = G(\bar{x}, \bar{a})$,

Далее предложим методы адаптации параметров $\bar{a}$ передаточной функции отдельно взятого нейрона, используя МНК или рекуррентный метод наименьших квадратов.

Пусть

$$g(y, \bar{a}) = \sum_{i=m+1}^{N} a_i g_i(y) \qquad (1)$$

$$f(\bar{x},\bar{a}) = \sum_{j=1}^{m} a_j f_j(\bar{x}_j) \tag{2}$$

Определим матрицу

$$H = \begin{bmatrix} -f_1(\bar{x}_1) & \dots & -f_m(\bar{x}_1) & g_{m+1}(y_1) & \dots & g_N(y_1) \\ \dots & \dots & \dots & \dots & \dots & \dots \\ -f_1(\bar{x}_k) & \dots & -f_m(\bar{x}_k) & g_{m+1}(y_k) & \dots & g_N(y_k) \end{bmatrix}$$

где $\bar{x}_l$ - вектор значений входов нейрона в *l-й* момент времени,

y_l – значения выхода нейронов в *l-й* момент времени.

$\bar{a}^T = [a_1 ... a_N]$

$\bar{E}^T = [1...1]$

$E = H^T \bar{a} + \xi$

Пусть $a_i > 0, \quad i = \overline{m+1, N}$

$\dfrac{\partial g_i(y)}{\partial y} > 0 \quad (**)$

Тогда (*) будет иметь единственное решение *y*.

Решая оптимизационную задачу, найдем оценки параметров $\hat{a}$ передаточной функции нейрона:

$$\begin{cases} \min\limits_{\bar{a}} (\bar{E} - H^T \bar{a})(\bar{E} - H^T \bar{a})^T, & (3) \\ a_i > 0, \quad i \equiv \overline{m+1, N} & (4) \\ g_i(y) > 0, \quad \forall y & (5) \end{cases}$$

Далее, используя оценки параметров $\hat{a}$, можем построить уравнение неявно заданной передаточной функции

$g(y, \hat{a}) = f(\bar{x}, \hat{a}) + 1$

Тогда для полученных значений параметров $\hat{a}$ существует единственное решение:

$y = G(\bar{x}, \bar{a})$

согласно лемме 1 при уловиях (4) и (5).

Решение $y = G(\bar{x}, \bar{a})$ является искомой передаточной функциейнейрона с адаптированными апраметрами $\hat{a}$.

Передаточную функцию можно получить как аналитически так и численно в зависимости от вида g и f.

Для случая, когда функции f и g заданы в виде (1) и (2), оценки параметров $\hat{a}$ можно получить по методу наименьших квадратов.

Принцип неявно заданных передаточных функций целесообразно использовать в самоорганизующихся сетях МГУА. Также возможно использование данного принципа как в традиционных сетях типа back propagation (в этом случае необходимо дополнительно вычислить $\frac{\partial y}{\partial a}$ для неявно заданной передаточной функции) так и в иных моделях нейронных сетей.

Использование принципа неявно заданных передаточных функций практически не отличается от использования явно заданных ПФ для метода группового учета аргумента, поэтому, данный пункт детализировать не будем.

Подробнее остановимся на использовании НПФ (неявно заданных передаточных функций) для сетей back propagation.

Рассмотрим градиентный метод обучения.

Пусть требуется минимизировать следующий критерий:

$$E(a) = \frac{1}{2}\sum_{p}\sum_{j}(y_{pj}^{N} - d_{pj})^2$$

где y_{pj}^{N}- реальный выход j-го нейрона выходного слоя N нейронной сети при подаче на вход p-го образа; d_{pj} - желаемый выход. Минимизируем критерий методом градиентного спуска:

$$a_{ij}(t) = a_{ij}(t-1) + \Delta a_{ij}$$

$$\Delta a_{ij}{}^{(N)} = -\eta\frac{\partial E}{\partial a_{ij}}$$

где a_{ij} – i-$й$ параметр j-го нейрона в слое.

$$\frac{\partial E}{\partial a_{ij}} = \frac{\partial E}{\partial y_j}\frac{\partial y_j}{\partial a_{ij}},$$

где $\dfrac{\partial y_j}{\partial a_{ij}} = \dfrac{\dfrac{\partial f}{\partial a_{ij}} - \dfrac{\partial g}{\partial a_{ij}}}{\dfrac{\partial g}{\partial y_j}}$

для g и f, заданных в виде (1) и (2) соответственно

$$\frac{\partial g}{\partial a_{ij}} = \begin{cases} g_i(y_j^n), & i > m \\ 0, & \backslash \end{cases}$$

где K- количество нейронов в предыдущем слое n-1.

n- текущий слой.

$$\frac{\partial f}{\partial a_{ij}} = \begin{cases} 0, & i > m \\ f_i(y_1^{(n-1)}, \dots, y_K^{(n-1)}), & \backslash \end{cases}$$

$$\frac{\partial g}{\partial y_j} = \sum_{i=m+1}^{N} a_i \frac{g_i(y_j)}{\partial y_j}$$

Далее распишем множитель $\dfrac{\partial E}{\partial y_j}$ как:

$$\frac{\partial E}{\partial y_j^{(n)}} = \sum_k \frac{\partial E}{\partial y_k^{(n+1)}} \frac{\partial y_k^{(n+1)}}{\partial y_j^{(n)}},$$

где $\dfrac{\partial y_k^{(n+1)}}{\partial y_j^{(n)}} = \dfrac{\sum\limits_{i=1}^{m} a_i \dfrac{f_i(y_j)}{\partial y_j^{(n)}}}{\sum\limits_{i=m+1}^{N} a_i \dfrac{\partial g_i}{\partial y_k^{(n+1)}}}$.

Если обозначить

$$\delta_j^{(n)} = \frac{\partial E}{\partial y_j^{(n)}}$$

Получаем рекуррентную формулу подсчета величины $\delta_j^{(n)}$ n-го слоя через последующий n+1-й слой:

$$\delta_j^{(n)} = \sum_k \delta_k^{(n+1)} \frac{\sum_{i=1}^{m} a_i \dfrac{f_i(y_j)}{\partial y_j^{(n)}}}{\sum_{i=m+1}^{N} a_i \dfrac{\partial g_i}{\partial y_k^{(n+1)}}}$$

$$\delta_j^{(N)} = (y_i^{(N)} - d_i)$$

И для простейшего случая градиентного метода получаем корректировку параметров сети (весов):

$$\Delta a_{ij}{}^{(n)} = -\eta \delta_j^{(n)} \frac{\partial y_j^{(n)}}{\partial a_{ij}^{(n)}}$$

Использование НПФ позволяет получить новые свойства нейронных сетей и вместе с тем ставит ряд новых задач для дальнейшего исследования искусственных нейронных сетей.

Список источников

1. D.J. Amit and H. Gutfreund. Spin-glass models of neural networks. *Physical Review A*, 32:1007 - 1018, 1985.

2. Олександрук Б.О. "Одношаговые методы обучения нейронных сетей аппроксимирующих элементов". АВТОМАТИКА 2001,

3. G. Carpenter and S. Grossberg. A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics, and Image Processing*, 37:54 - 115, 1987.

4. R.O. Grondin, W. Porod, C.M. Loeffler, and D.K. Ferry. Synchronous and asynchronous systems of threshold elements. *Biological Cybernetics*, 49:1 - 7, 1983.