зования естественных аналогов при создании моделей, технологий, методик, алгоритмов для решения тех или иных задач. Во многих случаях использование естественных аналогов дает положительные результаты. Это объясняется тем, что аналог, взятый из природы, совершенствовался в течение многих лет эволюции и имеет на данный момент самую совершенную в своем роде структуру.

Генетические алгоритмы – это мощная стратегия выхода из локальных оптимумов. Она заключается в параллельной обработке множества альтернативных решений, концентрируя поиск на наиболее перспективных из них.

В рассматриваемых программных средствах принята единая структура данных в виде совокупности массивов и файлов, содержащих описание схемы и результаты построения и анализа на полноту теста для нее. Наличие инвариантного ядра в данных средствах, включающего универсальную структуру данных и единый интерфейс, позволяет осуществлять их развитие путем модификации имеющихся решающих программных модулей и подключения новых.

На базе данных программных средств выполняется научно-исследовательская работа студентов, позволяющая получить практические навыки автоматизированного построения тестов.

УДК 004.021:032.26

Л. П. МАХНИСТ, И. И. ГЛАДКИЙ, Т. И. КАРИМОВА
Брест, БрГТУ

К ВОПРОСУ ОБУЧЕНИЯ МНОГОСЛОЙНЫХ НЕЙРОННЫХ СЕТЕЙ

В работе рассматривается искусственная нейронная сеть прямого распространения.

Обучение такой нейронной сети с использованием метода обратного распространения ошибки состоит в нахождении весовых коэффициентов w_{ij} и порогов T_j нейронной сети, которые минимизируют функцию ошибки сети, полученную по методу наименьших квадратов:

$$E(w_{11}, w_{21}, \ldots, w_{m1}, T_1, w_{12}, w_{22}, \ldots, w_{m2}, T_2, \ldots, w_{1n}, w_{2n}, \ldots, w_{mn}, T_n) = \frac{1}{2}\sum_{j=1}^{n}(y_j - t_j)^2,$$

где $y_j = F(S_j)$ – значение гладкой функции активации j-го выходного нейрона сети, $S_j = \sum_{i=1}^{m} w_{ij} x_i - T_j$, x_i – выходное значение i-го нейрона предыдущего слоя, t_j – ожидаемый выход j-го выходного нейрона ($i = \overline{1, m}$, $j = \overline{1, n}$).

Будем использовать обозначения:

$\overline{W} = (w_{11}, w_{21}, \ldots, w_{m1}, T_1, w_{12}, w_{22}, \ldots, w_{m2}, T_2, \ldots, w_{1n}, w_{2n}, \ldots, w_{mn}, T_n)^T$ – вектор-столбец весовых коэффициентов w_{ij} и порогов T_j нейронной сети, а $\overline{W}_j = (w_{1j}, w_{2j}, \ldots, w_{mj}, T_j)^T$ – вектор-столбец весовых коэффициентов w_{ij} и порога T_j, связанных с j-м выходным нейроном сети, $E(\overline{W}) = \sum_{j=1}^{n} E(\overline{W}_j)$ – функция ошибки сети, $E(\overline{W}_j) = \frac{1}{2}(y_j - t_j)^2$ – функция ошибки j-го выходного нейрона сети.

Изменение весовых коэффициентов w_{ij} и порогов T_j нейронной сети на каждом шаге обучения $(t+1)$ $(t = 1, 2, \ldots)$ производится по формулам:

$$\overline{W}_j(t+1) = \overline{W}_j(t) - \alpha(t) \nabla E(\overline{W}_j(t)), \tag{1}$$

где $\nabla E(\overline{W}_j(t))$ – градиент функции ошибки сети j-го выходного нейрона $E(\overline{W}_j(t))$ (например, в [1–4]).

В [5–8] рассматривались различные подходы к выбору шага обучения нейронной сети $\alpha(t)$, производился их сравнительный анализ с точки зрения сходимости алгоритма обучения с использованием метода наискорейшего спуска.

Предлагается после вычисления весовых коэффициентов w_{ij} и порогов T_j последнего слоя вычислять ожидаемые выходы нейронов предыдущего слоя $t_i = x_i$, которые минимизируют ошибку сети, как функцию, зависящую от значений x_i: $E(\overline{X}) = E(x_1, x_2, \ldots, x_m) = \frac{1}{2}\sum_{j=1}^{n}(y_j - t_j)^2$.

Обозначим через $\overline{X} = (x_1, x_2, \ldots, x_m)^T$ – вектор-столбец выходных значений нейронов предыдущего слоя.

Вычисление ожидаемых выходов нейронов предыдущего слоя $t_i = x_i$ будем производить с использованием метода наискорейшего спуска на каждом шаге $(t+1)$ $(t = 1, 2, \ldots)$ в соответствии с формулой:

$$\overline{X}(t+1) = \overline{X}(t) - \beta(t)\nabla E\left(\overline{X}(t)\right), \qquad (2)$$

где $\nabla E\left(\overline{X}(t)\right)$ – градиент функции ошибки сети $E\left(\overline{X}(t)\right)$.

При выполнении некоторых условий, приведенных, например, в [9], предлагается шаг $\beta(t)$ вычислять по формуле:

$$\beta(t) = \frac{\left(\nabla E\left(\overline{X}(t)\right), \nabla E\left(\overline{X}(t)\right)\right)}{\left(\nabla^2 E\left(\overline{X}(t)\right) \cdot \nabla E\left(\overline{X}(t)\right), \nabla E\left(\overline{X}(t)\right)\right)}, \qquad (3)$$

где $\nabla^2 E\left(\overline{X}(t)\right) \cdot \nabla E\left(\overline{X}(t)\right)$ – произведение матрицы Гессе $\nabla^2 E\left(\overline{X}(t)\right)$ функции $E\left(\overline{X}(t)\right)$ и вектора градиента $\nabla E\left(\overline{X}(t)\right)$, а $\left(\nabla E\left(\overline{X}(t)\right), \nabla E\left(\overline{X}(t)\right)\right)$ и $\left(\nabla^2 E\left(\overline{X}(t)\right) \cdot \nabla E\left(\overline{X}(t)\right), \nabla E\left(\overline{X}(t)\right)\right)$ – скалярные произведения соответствующих векторов.

С помощью формул (2) и (3) определяются ожидаемые выходы нейронов предыдущего слоя $t_i = x_i$. Далее в соответствии с формулой (1) предлагается производить изменение весовых коэффициентов и порогов предыдущего слоя. Следует заметить, что при таком подходе к обучению нейронной сети целесообразно использовать функции активации, множество значений которых есть вся числовая прямая.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Гладкий, И. И. Обучение нейронных сетей с использованием метода наискорейшего спуска / И. И. Гладкий, В. А. Головко, Л. П. Махнист // Вестн. Брест. гос. техн. ун-та. Сер.: Физика, математика, информатика. – 2001. – № 5. – С. 47–55.

2. Maniakov, N. Traing algorithm for forecasting multilayer neural network / N. Maniakov, L. Makhnist, V. Rubanov // Pattern Recognition and Information Processing : Proceedings of The Seventh International Conferences (PRIP'2003), Minsk, Republic of Belarus, 21–23 May 2003 : in 2 vol. – Minsk, 2003. – Vol. 1. – P. 26–30.

3. Golovko, M. Multilayer neural networks training methodic / M. Golovko, L. Makhnist, N. Maniakov // Second IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Ap-

4. Makhnist, L. Some Methods of Adaptive Multilayer Neural Network Training / L. Makhnist, N. Maniakov // International Journal of Computing. – 2004. – Vol. 3. – P. 99–106.

5. Maxnist, L. Convergence Analysis of Neural Networks Training Based on steepest Descent Method / L. Maxnist, A. Doudkin, V. Golovko // Pattern Recognition and Information Processing (PRIP'2007) : Proceedings of the Ninth International Conference, Minsk, Republic of Belarus, 22–24 May 2007 : in 2 vol. – Minsk, 2007. – Vol. 1. – P. 285–289.

6. Гладкий, И. И. О выборе шага обучения искусственных нейронных сетей прямого распространения / И. И. Гладкий, Л. П. Махнист, В. С. Рубанов // Вычислительные методы, модели и образовательные технологии : сб. материалов VIII Междунар. науч.-практ. конф., Брест, 18 окт. 2019 г. / Брест. гос. ун-т им. А. С. Пушкина ; под общ. ред. А. А. Козинского. – Брест : БрГУ, 2019. – С. 121–122.

7. Махнист, Л. П. О сходимости алгоритмов обучения искусственных нейронных сетей / Л. П. Махнист, А. В. Санюкевич, В. П. Черненко // Вычислительные методы, модели и образовательные технологии : сб. материалов VIII Междунар. науч.-практ. конф., Брест, 18 окт. 2019 г. / Брест. гос. ун-т им. А. С. Пушкина ; под общ. ред. А. А. Козинского. – Брест : БрГУ, 2019. – С. 133–135.

8. Махнист, Л. П. Оценки скорости сходимости и выбор шага обучения искусственных нейронных сетей прямого распространения / Л. П. Махнист, В. А. Головко, И. И. Гладкий // Вестн. Брест. гос. техн. ун-та. Сер.: Физика, математика, информатика. – 2019. – № 5. – С. 27–35.

9. Махнист, Л. П. Использование алгоритмов обучения однослойной сети для многослойных нейронных сетей прямого распространения / Л. П. Махнист, В. А. Головко, И. И. Гладкий // Вестн. Брест. гос. техн. ун-та. Сер.: Физика, математика, информатика. – 2020. – № 5. – С. 32–37.

УДК 004.942:519.218

Л. П. МАХНИСТ, Е. Н. ЗАЩУК, И. И. ГЛАДКИЙ
Брест, БрГТУ

К РЕШЕНИЮ ЗАДАЧИ ГИДРОЛОГИИ С ИСПОЛЬЗОВАНИЕМ СИСТЕМ КОМПЬЮТЕРНОЙ АЛГЕБРЫ

Рассматривается стохастическая модель многолетних колебаний речного стока, которую можно записать в виде системы дифференциальных уравнений с граничными условиями (например, в [1] и [2]):

$$\frac{d^2\theta_k}{d\xi^2} - \xi\frac{d\theta_k}{d\xi} = -k\theta_{k-1}, \text{ при } \frac{d\theta_k}{d\xi}(+\infty) = 0, \ \theta_k(\xi)\big|_{\xi=\xi_*} = 0 \ \ (\theta_0 = 1, \ k \in N). \quad (1)$$