Е. В. П у ч к о в, **В. Б. Л и л а** (Ростов-на-Дону, РГСУ). **Исследование алгоритма обучения многослойного персептрона, в котором используются методы сопряженных градиентов.**

В работе, представленной данным сообщением, рассматривается сочетание метода обратного распространения ошибки [1] и методов сопряженных градиентов [2].

Спуск по сопряженным градиентам — это методы оптимизации, хорошо зарекомендовавшие себя при поиске условного и безусловного экстремума. К этим методам относятся методы, связанные с запоминанием направлений спуска или подъема на предыдущих шагах.

Несмотря на то, что наилучшим направлением спуска или подъема является антиградиент или градиент, для широкого класса целевых функций эти градиентные методы не применимы.

В ситуациях, когда градиентные методы не приводят к успеху, методы сопряженных градиентов успешно справляются с поставленной задачей.

В связи с этим возникает интерес разработки алгоритма обучения искусственной нейронной сети, в основе которого заложены методы сопряженных градиентов.

Обучение многослойного персептрона заключается в отыскании вектора W, для которого суммарная ошибка по всем примерам (опытам) минимальна:

$$E = \sum_{n=1}^{N} e_n = \min . \tag{1}$$

Обычно для ошибки по одному примеру применяют формулу $e_n = (1/2)\|y_n - f(w, x_n)\|^2$.

Как и всякий алгоритм обучения ИНС, исследуемый алгоритм является итерационным. Итерацией будем считать одну эпоху, т. е. когда будут поданы сети все обучающие примеры.

Описание алгоритма.

1. Инициализировать сеть. Назначить начальные значения параметров — стартовую точку для весовых коэффициентов W_0, начальное направление движения d_0 и начальный шаг $\eta_{(0)}\, d_{(0)} = g_{(0)} = -F'(w_0)$.

2. Выбрать очередной вектор из обучающего множества и подать его на вход сети.

3. Вычислить выход сети $y(x)$по формуле $y(x) = \sum_{i=1}^{N} x_i w_{ij}$.

4. Вычислить разность между выходом сети и требуемым значением для данного вектора (1).

5. Если была допущена ошибка при классификации выбранного вектора, то необходимо подкорректировать вектор весов как линейную комбинацию предыдущего вектора и текущего вектора направления: $W(n+1) = W(n) + \eta(n)d(n)$.

6. На очередном шаге определить направление движения $d(n)$ по формуле $d(n) = g(n) + \beta(n)\, d(n-1)$ на каждом слое. Если $\beta(n)$ вычисляется по формуле

$$\beta(n) = [g^T(n+1)g(n+1)]/[g^T(n)g(n)], \tag{2}$$

то получаем метод Флетчерса–Ривса. Отметим, что $\beta(n)$ можно рассчитывать еще 4 способами, получая тем самым различные модификации алгоритма сопряженных градиентов.

7. Определить $\eta(n)$ из условия $E(W(n) + \eta(n)d(n))$ для каждого слоя. Для этого можно воспользоваться методом Фибоначчи, методом золотого сечения или методом дихотомии. Более быструю сходимость обеспечивает метод Ньютона–Рафсона, но для этого необходимо иметь возможность вычисления матрицы Гессе.

8. Если ошибка недостаточно мала и не выполнено какое-либо правило останова, то шаги 2 и 7 повторить. Чтобы осуществить рестарт алгоритма, т. е. забыть последнее направление поиска и запустить алгоритм заново в направлении скорейшего

спуска, в формуле (2) β присваивается 0 через каждые $n+1$ шагов, где n — число весов в настраиваемом слое.

В предлагаемом алгоритме используется свойство сходимости за фиксированное число шагов для квадратичной целевой функции. Метод опробован на серии задач, в которых он показал более высокую скорость сходимости по сравнению с методом обратного распространения ошибки.

СПИСОК ЛИТЕРАТУРЫ

1. *Хайкин С.* Нейронные сети. М.: Вильямс, 2006, 1104 с.
2. *Тархов Д. А.* Нейронные сети. Модели и алгоритмы. Кн. 18. Справочное издание (серия «Нейрокомпьютеры и их применение»). М.: Радиотехника, 2005, 256 с.