УДК 004.827

ВЫЧИСЛЕНИЕ ИНТЕРВАЛЬНЫХ ЧИСЕЛ НА ОСНОВЕ НЕЙРОННЫХ СЕТЕЙ ПРЯМОГО РАСПРОСТРАНЕНИЯ

С. С. Пархоменко, Т. М. Леденёва

Воронежский государственный университет

Поступила в редакцию 06.04.2015 г.

Аннотация. в статье предложен алгоритм обработки многомерных зашумлённых данных для аппроксимации функциональной зависимости с помощью нейронных сетей прямого распространения, предоставляющей на выходе интервальные числа.
Ключевые слова: нейронная сеть прямого распространения, метод Левенберга-Марквардта, интервальное число.
Annotation. This article reviews an algorithm of multidimensional noisy data processing by neural networks to get an approximate function returning intervals.
Keywords: neural networks, Levenberg-Marquardt method, intervals.

ВВЕДЕНИЕ

Предположим, что наблюдается некоторый параметр Y, зависящий от ряда факторов $X_1,...,X_n$, так что можно говорить о существовании функции $Y = f(X_1,...,X_n)$. С учетом обозначения $X = (X_1,...,X_n)$ получим $Y = f(X)$. Значения переменных $Y, X_1,...,X_n$ будем обозначать соответствующими малыми буквами. Нередко оказывается, что функция f участвует в физико-технических или математических расчетах, где ее приходится многократно вычислять. Если вид функции неизвестен или вычисления являются дорогостоящими, то целесообразно заменить функцию f приближенной формулой, т. е. на основе наблюдаемых данных $\left\{\left(\boldsymbol{x}^i = \left(x_1^i,...,x_n^i\right), y_i\right)\right\}_{i=\overline{1,N}}$ найти такую функцию $F(X)$, которая близка в некотором смысле к $f(X)$. При всех значениях векторного аргумента $\boldsymbol{x}$ предположим, что $f(\boldsymbol{x}) \approx F(\boldsymbol{x})$, тогда функция F называется аппроксимирующей.

Приведенная постановка задачи хорошо известна как задача приближения (в данном случае аппроксимации) функции [1]. Для ее решения существует значительное количество методов, при этом различают среднеквадратическое приближение (частный случай – метод наименьших квадратов) и равномерное приближение, различающиеся критериями близости $F(X)$ и $f(X)$.

В рамках сформулированной выше задачи введем ряд предположений, которые значительно усложняют постановку и делают неприменимыми классические методы приближения, а, следовательно, требуют разработки новых подходов.

1. Каждый фактор $X_k \left(k = \overline{1,N}\right)$ измеряется с погрешностью, так что его значением будет величина $x_k + \varepsilon_k$, где ε_k – ошибка, т. е. реализация случайной величины, распределенной по закону $N\left(0,\sigma^2\right)$. Фиксирование параметра Y также производится с погрешностью $\varepsilon_Y(\boldsymbol{x})$.

2. Имеет место многозначное отображение множества значений входного вектора X во множество значений наблюдаемого параметра Y, при этом конкретному значению x_i фактора X_i соответствует множество значений наблюдаемого параметра Y, так что при анализе всей выборки оказывается, что x_i соответствует некоторый интервал $[y_i] = [\underline{y}_i, \overline{y}_i]$, содержащий наиболее вероятные результаты наблюдений Y при данном значении фактора x_i. Таким образом, вектору $\boldsymbol{x}^i = \left(x_1^i,...,x_n^i\right)$ значений факторов $X_1,...,X_n$ ставится в соот-

© Пархоменко С. С., Леденёва Т. М., 2015

ветствие интервал $[y_i]$, в качестве значения наблюдаемого параметра Y.

Под интервальным числом $[a]$ будем понимать вещественный отрезок $[\underline{a}, \overline{a}]$, где $\underline{a} \le \overline{a}$. Обычное число является частным случаем интервального при $\underline{a} = \overline{a} = a$. Для интервального числа $[a]$ можно рассматривать две характеристики: ширину $\mathrm{wid}([a]) = \overline{a} - \underline{a}$ и середину $\mathrm{med}([a]) = \frac{\underline{a} + \overline{a}}{2}$ [2].

Таким образом, выборка представляет собой множество вида

$$D = \left\{ \left(\boldsymbol{x}^i, y_i \right) \right\}_{i=\overline{1,N}}, \qquad (1)$$

на основе которой, с помощью функции $F(X)$ необходимо получить интервальные числа $[y_i]$. В данной формулировке задачи классические методы приближения оказываются неприменимыми.

Для работы со сложными многомерными данными можно воспользоваться нейронными сетями прямого распространения (НСПР) – структурой, состоящей из следующих друг за другом слоёв нейронов. В качестве входов каждый последующий слой использует выходы предыдущего слоя (обычно называемого скрытым слоем) за исключением входного слоя, непосредственно принимающего входные сигналы [3]. Основная функция скрытых слоёв НСПР – выделение статистики высокого порядка. НСПР, по сути, является универсальным аппроксиматором и, обладая фиксированной архитектурой, в состоянии обучаться и воспроизводить (восстанавливать) сложные гиперповерхности. В основе функционирования НСПР лежит теорема Фунахаши (Funahashi) [4], согласно которой с помощью (как минимум двухслойной) НСПР можно аппроксимировать любую непрерывную функцию, определенную на компакте, с любой заданной точностью.

В общем случае при наличии обучающего множества (выборки) $\left\{ \left(\boldsymbol{x}_i, \boldsymbol{y}_i \right) \right\}_{i=\overline{1,N}}$, где $\boldsymbol{x}_i \in R^n$ – входной вектор i-го примера, n – количество входов НСПР, $\boldsymbol{y}_i \in R^m$ – вектор ожидаемых значений (указателя учителя), m – количество выходов сети, k – количество её настраиваемых параметров (весов и смещений), целью обучения НСПР является определение такого вектора весов w^*, чтобы

$$w^* = \arg\min_{w \in R^k} E_D = \\ = \arg\min_{w \in R^k} \sum_{i=1}^{N} \frac{1}{2} \left(F\left(\boldsymbol{x}_i, w \right) - \boldsymbol{y}_i \right)^2. \qquad (2)$$

От обученной НСПР ожидается способность описывать зависимость в данных, не включенных в обучающее множество, т. е. сеть должна обеспечивать обобщение примеров обучающего множества [1]. Такая способность сети называется обобщающей. В случае «плохого» обучения низкий уровень обобщающей способности влияет на неустойчивое поведение НСПР, выражающееся в появлении на выходе маловероятных и/или некорректных значений, в результате чего она сама становится дополнительным источником некорректных данных.

1. РЕГУЛЯРИЗАЦИЯ КАК СРЕДСТВО ДЛЯ ПОВЫШЕНИЯ УРОВНЯ ОБОБЩАЮЩЕЙ СПОСОБНОСТИ НСПР

К основным методам поддержки обобщающей способности НСПР относятся [1]: кросс-валидация (скользящий контроль, перекрёстная проверка) и регуляризация сложности.

Идея кросс-валидации заключается в разбиений исходного множества D на два подмножества: обучающего и контрольного. С помощью последнего осуществляется контроль за несмещённой оценкой вероятности ошибки [5].

Под *регуляризацией* будем понимать метод добавления некоторой дополнительной информации к условию с целью предотвращения переобучения.

При регуляризации стандартная функция обучения E_D из (2) дополняется функцией штрафа за избыточную сложность:

$$E_R = E_D + \theta E_C, \qquad (3)$$

где $E_{\tilde{N}}$ – функция штрафа за сложность, θ – параметр регуляризации, который характеризует относительную значимость E_C относительно E_D. В качестве вариантов задания E_C можно рассматривать следующие [1, 6]:

а) $E_C = \|w\|^2$;

б) $E_C = \sum_{i=1}^{W} \frac{(w_i/w_0)^2}{1+(w_i/w_0)^2}$, где w_0 – некоторой предопределённый параметр, w_i – вес i-го синапса НСПР;

в) $E_C = \sum_{i=1}^{q} w_{oi}^2 \left\|\overline{w}^{(i)}\right\|^p$, где w_{oi} – вес выходного слоя, идущий от i-го нейрона скрытого слоя; $\overline{w}^{(i)}$ – вектор весов i-го нейрона скрытого слоя; p – порядок сглаживания, определяемый правилом: $p = 2s$ – для локального сглаживания или $p = 2s-1$ – для глобального сглаживания (в обоих случаях параметр s определяется порядком дифференцирования функции $F(\boldsymbol{x}_i, w)$.

Выбор функции обучения E_C зависит от структуры НСПР и допустимой вычислительной сложности.

Одним из способов обучения НСПР является байесовский подход [7 8], согласно которому суть обучения сводится к максимизации вероятности порождения различными векторами весовых коэффициентов w обучающего множества D, при этом вектор весов определяется как решение задачи:

$$w^* = \arg\max_{w} P(w|D), \qquad (4)$$

где w_{MP} – наиболее вероятные весовые коэффициенты.

Пусть существуют различные варианты моделей (структур) $\{H_i\}_{i\in I}$ нейронной сети, тогда для фиксированной модели H_i выбор вектора весов w определяется формулой Байеса

$$P(w|D,H_i) = \frac{P(D|w,H_i)P(w|H_i)}{P(D|H_i)}, \qquad (5)$$

где $P(w|D,H_i)$ – апостериорная вероятность получения вектора весов w при условии, что НСПР имеет структуру H_i и используется обучающее множество D; $P(D|w,H_i)$ – вероятность того, что НСПР с вектором весов w «породит» обучающее множество D; $P(w|H_i)$ – априорная вероятность выбора w в рамках модели H_i; $P(D|H_i)$ – вероятность выбора обучающего множества D для данной НСПР.

Формула (6) позволяет для вектора весов w определить распределение вероятностей, а, следовательно, ввести в рассмотрение случайную величину W.

В предположении (на основе центральной предельной теоремы), что распределение вероятностей данной случайной величины аппроксимируется нормальным законом с параметрами $a = w^*$ и $\sigma = \Delta w$, в [7] показано, что

$$P(D|H_i) =$$
$$= P(D|w^*,H_i)P(w^*|H_i)(2\pi)^{k/2}(\det A)^{-1/2},$$

где A – матрица гессиана для $\log P(w|D,H_i)$.

В [7, 9] предложены следующие модифицированные формулы для перечисленных вероятностей в рамках предположений о нормальности соответствующих распределений с учетом дополнительных параметров α и β:

$$P(w|D,\alpha,\beta,H_i) = \frac{P(D|w,\beta,H_i)P(w|a,H_i)}{P(D|a,\beta)}, \qquad (6)$$

$$P(D|w,\beta,H_i) = \frac{\exp\left(-\beta E_D(D|w,H_i)\right)}{(2\pi/\beta)^{N/2}}, \qquad (7)$$

$$P(w|\alpha,H_i) = \frac{\exp\left(-\alpha E_W(w|H_i)\right)}{(2\pi/\alpha)^{N/2}}, \qquad (8)$$

$$P(D|\alpha,\beta,H_i) \approx$$
$$\approx \frac{(2\pi)^{N/2} e^{-F(w^*,H_i)}\left(\det\left(\beta\nabla^2 E_D + \alpha\nabla^2 E_W\right)\right)^{-1/2}}{(2\pi/\alpha)^{k/2}(2\pi/\beta)^{N/2}}, \qquad (9)$$

где $E_D(D|w,H_i)$ – квадратичная функция, характеризующая отклонения предсказанных значений выходной переменной от значений этой же переменной в обучающем множестве; $E_W(w|H_i) = \frac{1}{2}\sum_i w_i^2 = \frac{1}{2}\|w\|^2$; β –мера предполагаемого шума в обучающем множестве D, α – параметр регулярной составляющей.

Оптимальные весовые коэффициенты должны максимизировать вероятность $P(w|D,\alpha,\beta,H_i)$, что достигается за счет минимизации комбинированной целевой функции [7, 9 10]:

$$F_B = \beta E_D + \alpha E_W. \qquad (10)$$

Параметры α и β конкретизируют способ оптимизации функции F_B: либо с помо-

щью минимизации смещений весовых коэффициентов, либо за счёт минимизации ошибки обучения. Их точные значения зависят от текущего состояния НСПР и рассчитываются по следующим формулам [7, 9 10]:

$$\alpha = \frac{\gamma}{2E_W},\ \beta = \frac{N-\gamma}{2E_D}, \qquad (11)$$

где $\gamma = k - 2\alpha \cdot \operatorname{tr} H_B^{-1}$; k – количество настраиваемых параметров модели НСПР; $\operatorname{tr} H_B^{-1}$ – след обратной матрицы Гессе $H_B = \nabla^2 F_B$.

Для α существует модификация Поланда [10] вида

$$\alpha = \frac{k}{2E_W + \operatorname{tr} H_B^{-1}}, \qquad (12)$$

которую рекомендуется применять для небольших обучающих выборок. Для повышения численной допускается применение «отложенного» изменения коэффициента α, например, спустя 10 итераций обучения НСПР [10].

2. ОБУЧЕНИЕ НСПР МЕТОДОМ ЛЕВЕНБЕРГА-МАРКВАРДТА

Метод обучения НСПР влияет на способность выделять зависимости в зашумлённых данных. Распространенным методом обучения является метод обратного распространения ошибки. Несмотря на широкое применение, его главные недостатки – медленная сходимость и негативное влияние локальных минимумов поверхности E_D [1,3]. Эти недостатки преодолены в методе Левенберга-Марквардта, суть которого сводится к многократному решению уравнения [3]

$$\left(J^T J + \lambda I\right)\delta = J^T\left(\tilde{y} - F\left(\tilde{x}, w\right)\right), \qquad (13)$$

где δ – вектор, состоящий из величин приращения весов;

$$J = \begin{bmatrix} \dfrac{\partial F\left(x_1, w\right)}{\partial w_1} & \cdots & \dfrac{\partial F\left(x_1, w\right)}{\partial w_W} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial F\left(x_N, w\right)}{\partial w_1} & \cdots & \dfrac{\partial F\left(x_N, w\right)}{\partial w_W} \end{bmatrix}. \qquad (14)$$

Регуляризация Байеса органично сочетается с методом Левенберга-Марквардта, модифицируя формулу (13) следующим образом [11]:

$$\left(2\beta J^T J + 2I\left(\alpha + \lambda\right)\right)\delta = J^T\left(\tilde{y} - F\left(\tilde{x}, w\right)\right), \qquad (15)$$

где

$$H \approx J^T J, \quad H_B \approx 2\beta J^T J \approx 2\beta H. \qquad (16)$$

Выделив с помощью НСПР общую зависимость необходимо проанализировать соответствие исходных данных этой модели. Если принять, что обучающая выборка корректна и причиной шума является недостаток данных об исследуемом объекте и/или изначально заложенная в него неопределённость, то из шума необходимо извлечь дополнительную информацию и агрегировать её, например, в виде *интервала предсказаний*. Он рассчитывается на основе исходных данных и определяет интервал, в котором с известной вероятностью должны появиться новые наблюдения, если способ их получения остаётся прежним [12]. Для этого предположим, что шум представляет собой независимые одинаково распределённые случайные величины [13].

Одним из самых простых, но эффективных методов вычисления интервалов предсказаний является дельта-метод [13 14], в рамках которого интервал предсказаний определяется следующим образом:

$$\hat{y}_0 \pm t_{N-k-1}^{1-\frac{\alpha}{2}} \hat{\sigma}_\varepsilon \sqrt{1 + g_0^T H^{*-1} g_0}, \qquad (17)$$

где $g_0^T = \left[\dfrac{\partial F\left(x_0, w^*\right)}{\partial w_1^*}, \ldots, \dfrac{\partial F\left(x_0, w^*\right)}{\partial w_p^*}\right]$;

$\hat{\sigma}_\varepsilon = \dfrac{1}{n-1}\sum_{i=1}^{n}\left(F\left(\tilde{x}_i, w\right) - \tilde{y}_i\right)^2$; $t_{N-W-1}^{1-\frac{\alpha}{2}}$ – квантиль распределения Стьюдента уровня $\left(1 - \dfrac{\alpha}{2}\right)$ со степенью свободы $\left(N - k - 1\right)$, w^* – весовые коэффициенты обученной НСПР, x_0 – входной вектор, H^{*-1} – матрица Гессе, соответствующая весовым коэффициентам обученной НСПР, которую можно извлечь из (15) и (16) на последнем шаге обучения.

Для повышения точности вычисления интервалов предсказания предлагается применить метод максимального правдоподобия к $\hat{\sigma}_\varepsilon$ и принять $\hat{\sigma}_\varepsilon \approx F_E\left(\boldsymbol{x}, w^*\right)$, где $F_E\left(\boldsymbol{x}, w^*\right)$ – функция аппроксимации обучающего множества $\left\{\left(\boldsymbol{x}^i, \left|F\left(\boldsymbol{x}^i, w\right) - y_i\right|\right)\right\}_{i=\overline{1,N}}$, которую можно вычислить с помощью дополнительной НСПР [14].

3. ОПИСАНИЕ АЛГОРИТМА

На основе проведенных рассуждений разработан механизм обработки данных на основе НСПР, обученной методом Левенберга-Марквардта, с применением регуляризации Байеса и вычислением интервала предсказания с использованием комбинации дельта-метода и метода максимального правдоподобия. Механизм представим в виде комбинации трёх алгоритмов: обучения НСПР, компоновки и расчёта интервальных чисел.

Алгоритм обучения НСПР:

1) Инициализировать $w_i = rand(-0.25, 0.25)$, $\alpha = 0$, $\beta = 1$, $H_{B^{\cdot\cdot}}^{-1} = 1$, $\lambda = 1000$, $s = 10$, $\upsilon = 10$, $i = 1$, $\psi \to +0$.

2) Если $\lambda < \psi$, то принять $\lambda = \psi$.

3) Последовательно вычислить J по формуле (14), $g = J^T\left(\tilde{y} - F\left(\tilde{x}, w\right)\right)$,

$$F_B^1 = \beta E_D + \alpha E_W,\ H_B \approx 2\beta J^T J.$$

4) Рассчитать $\left(H_B + 2I\left(\alpha + \lambda\right)\right)\delta = g$ относительно неизвестного δ.

5) Обновить весовые коэффициенты w с помощью δ.

6) Вычислить $F_B^2 = \beta E_D' + \alpha E_W'$ для нового вектора весов.

7) Если $F_B^1 < F_B^2$, то отклонить обновление вектора весов w, увеличить λ с помощью умножения на вспомогательную величину υ и перейти на шаг 7.

8) Уменьшить λ с помощью деления на υ.

9) Вычислить $\gamma = k - 2\alpha \cdot \operatorname{tr} H_B^{-1}$.

10) Если $i > s$, то вычислить

$$\alpha = \frac{k}{2E_W' + \operatorname{tr} H_B^{-1}}.$$

11) Вычислить $\beta = \dfrac{N - \gamma}{2E_D'}$.

12) Увеличить счетчик итераций i на 1. Если $i < i_{\max}$ и НЕ выполнено условие досрочной остановки Q, то перейти на шаг 2.

13) На последней итерации матрицу H, полученную на шаге 3 с учётом (16), сохранить как $H^{*-1} = H^{-1}$.

С помощью параметра s указывается количество итераций, на которое, благодаря условию в шаге 13, «откладывается» изменение α. Благодаря ψ и шагу 2 гарантируется, что в результате погрешностей вычислений не будет достигнуто $\lambda = 0$. Параметр $i_{\max}$ определяет максимальное количество итераций обучения НСПР. Параметр Q определяет условие досрочной остановки обучения до достижения $i = i_{\max}$, благодаря которому можно сократить время обучения НСПР и предотвратить вычислительные ошибки при достижении оптимума.

Для классического метода Левенберга-Марквардта чаще всего используется условие $Q_1 : \left|E_D^{i+1} - E_D^i\right| \le \varepsilon_1$, где ε_1 – некоторое малое значение, E_D^i – значение E_D на шаге i, которое с учётом регуляризации Байеса трансформируется в $\left|F_B^1 - F_B^2\right| \le \varepsilon_1$. Для более тонкого контроля за E_W можно использовать условие $Q_2 : \|\delta\| \le \varepsilon_2$ [6], причём ε_2 можно рассчитать, например, как $\varepsilon_2 = k\left(\Delta_{\min} w\right)^2$, где $\Delta_{\min} w$ – минимально допустимое изменение одного весового коэффициента НСПР. При использовании $Q_3 : \operatorname{tr} H_B^{-1} < \varepsilon_3$, где ε_3 – малое число, проверяется численная устойчивость алгоритма. Дополнительно необходимо избежать бесконечного роста $\left(\lambda + \alpha\right)$, возникающего в конце обучения из-за отсутствия возможности дальнейшей оптимизации по F_B, что приводит к введению условия $Q_{пр} : \lambda + \alpha > \varepsilon_{пр}$, где $\varepsilon_{пр}$ – достаточно большое число. Таким образом, общее условие досрочной остановки обучения НСПР Q имеет вид $Q : Q_{[1,2,3]} \vee Q_{пр}$.

Необходимо отметить, что выбор между Q_1, Q_2, Q_3 может зависеть от задачи, способа обучения и применения НСПР. Так, например, условие Q_2 используется при контроле за изменением весовых коэффициентов и подборе оптимальной структуры НСПР, при этом не гарантируется оптимальность ни F_B, ни $\|\delta\|$. Условие Q_3 позволяет добиться максимально точного обучения НСПР.

При выборе ψ, ε_1, ε_2 ($\Delta_{\min} w$), ε_3, $\varepsilon_{пр}$ необходимо учитывать точность вычислений, используемое аппаратное и программное обеспечение.

Изложим *алгоритм компоновки механизма*:

1) Обучить НСПР $F\left(\boldsymbol{x}, w\right)$ с помощью алгоритма обучения НСПР по нормализо-

ванной обучающей выборке D, при этом сохранить H^{*-1}.

2) Найти обучающую выборку

$$D' = \left\{\left(\boldsymbol{x}^i, \left|F\left(\boldsymbol{x}^i, w\right) - y_i\right|\right)\right\}_{i=\overline{1,N}}.$$

3) Обучить НСПР $G(\boldsymbol{x}, w)$ с помощью алгоритма обучения НСПР по нормализованной обучающей выборке D'.

Алгоритм вычисления интервального числа включает следующие шаги:

1) Задать входное значение $\boldsymbol{x}^0$.

2) Вычислить $y_0 = F\left(\boldsymbol{x}^0, w\right)$.

3) Вычислить $y_\Delta = t_{N-k-1}^{1-\frac{\alpha}{2}} z G(x_0, w)$, где $z = \sqrt{1 + g_0^T H^{*-1} g_0}$.

4) Определить интервальное число $[y_0] = [y_0 - y_\Delta, y_0 + y_\Delta]$ в качестве предсказанного значения выходной переменной y.

4. РЕЗУЛЬТАТЫ ВЫЧИСЛИТЕЛЬНОГО ЭКСПЕРИМЕНТА

Для проведения вычислительного эксперимента была разработана программа на языке C с применением технологии параллельного программирования OpenMP.

Пусть имеется некоторый алгоритм с вычислительной сложностью $\underline{o}(n) = n$, $\overline{O}(n) = n^2$, $n \in R$ и произведено несколько его запусков с различными n. Принимая, что время работы алгоритма $t \in \left[\underline{o}(n), \overline{O}(n)\right]$, необходимо определить, сколько будет потрачено времени на n_0. Используем для обучения две (одну для расчёта средней величины t, другую – для Δt) НСПР с 5 нейронами в скрытом слое. Положим $i_{\max} = 15000$, $Q = Q_3 \vee Q_{up}$, $t_{500-16-1}^{1-\frac{0.1}{2}} \approx 1.648$, $\psi = 1000 D_{\min}$, $\varepsilon_3 = 0.00001$, $\varepsilon_{up} = D_{\max}/1000000$, $s = 15$, где $D_{\min}$ и $D_{\max}$ – соответственно минимальное и максимальное абсолютное ненулевое значение допустимое в вычислениях (точные значения определены в системных библиотеках разработчика).

На рис. 1 изображён отклик предложенного механизма предсказания. Крестиком выделены элементы обучающей выборки. Видно, что механизм корректно достраивает как среднюю величину с помощью функции $F(\boldsymbol{x}, w)$ (штрихпунктирная линяя), так и интервалы (сплошные линии), благодаря

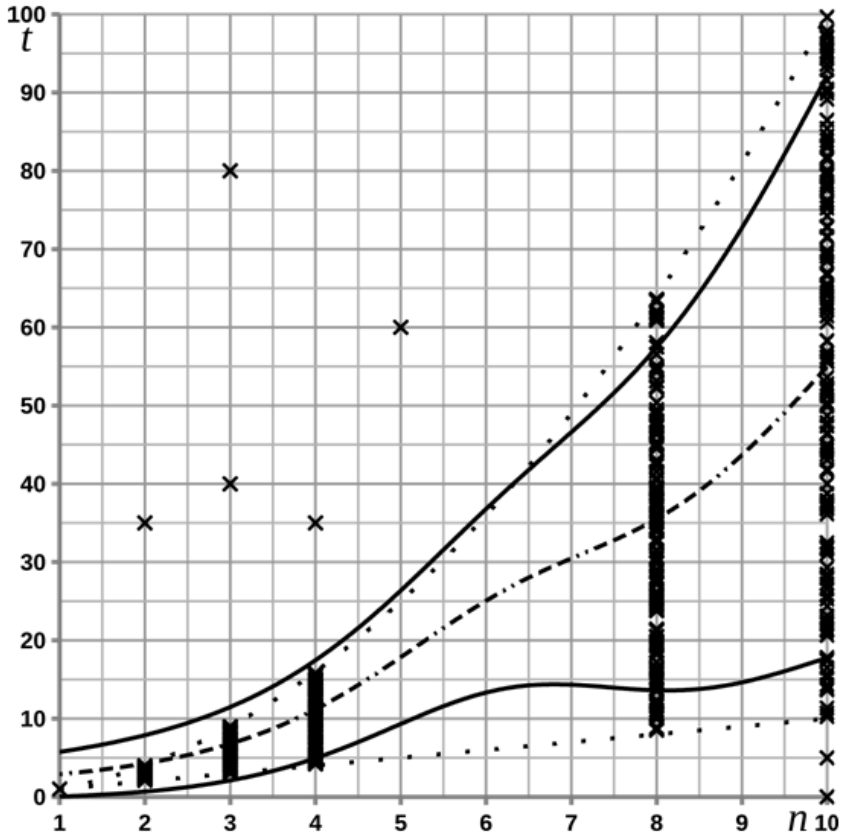

Рис. 1. Обучающая выборка и рассчитанные интервалы предсказания

$G(\boldsymbol{x}, w)$, в условиях отсутствия данных в этих областях (редким штрихом выделены границы интервала $\left[\underline{o}(n), \overline{O}(n)\right]$).

Рассмотрим другой пример: $z = x^2 + y^2$. Построим обучающую выборку из области $x, y \in [-2, 2)$ с помощью сетки с шагами $\Delta x, \Delta y = 0.05$. Добавим случайные величины $\varepsilon \in [-1, 1]$ к области $\Delta x, \Delta y \in [0.8, 1.2]$, с шагом $\Delta x_\varepsilon, \Delta y_\varepsilon = 0.1$. Обучим две НСПР с 5ю нейронами в скрытом слое при $i_{\max} = 25000$, $Q = Q_3 \vee Q_{up}$, $t_{6400-16-1}^{1-\frac{0.1}{2}} \approx 1.645$.

После обучения функция $F(\boldsymbol{x}, w)$ в обоих случаях корректно восстанавливает заданную поверхность. Поверхность ошибки $G(\boldsymbol{x}, w)$ различалась незначительно. Однако если представить Δy в виде поверхности, то заметим, что при отсутствии шума она является пологой. В случае зашумления исходных данных на поверхности формируются возмущения. Заметим, что в данном случае по рис. 2 фиксируется два пика в окрестностях $(0.5, 0.5)$ и $(1.4, 1.4)$. Если через эти две точки построить прямоугольную область, в неё попадёт область зашумленных данных, из-за чего окрестность возвышается над той частью поверхности, где шума нет.

В ходе проведения экспериментов установлено:

1) регуляризация Байеса необходима для поддержания обобщающей способности НСПР;

2) корректность обучения во многом зависит от объёма и разнообразия исходных данных: чем больше в некоторой области сосредоточено наблюдений, тем точнее механизм обрабатывает случайные выбросы и достраивает интервалы предсказания;

3) для получения максимально точного значения y_Δ рекомендуется увеличить количество итераций обучения;

4) при $s = 0$ наблюдалось некорректное завершение процесса обучения;

5) необходимо учитывать, что расчёт интервалов производится по результатам обучения НСПР (по функции $F(\boldsymbol{x}, w)$); исходная

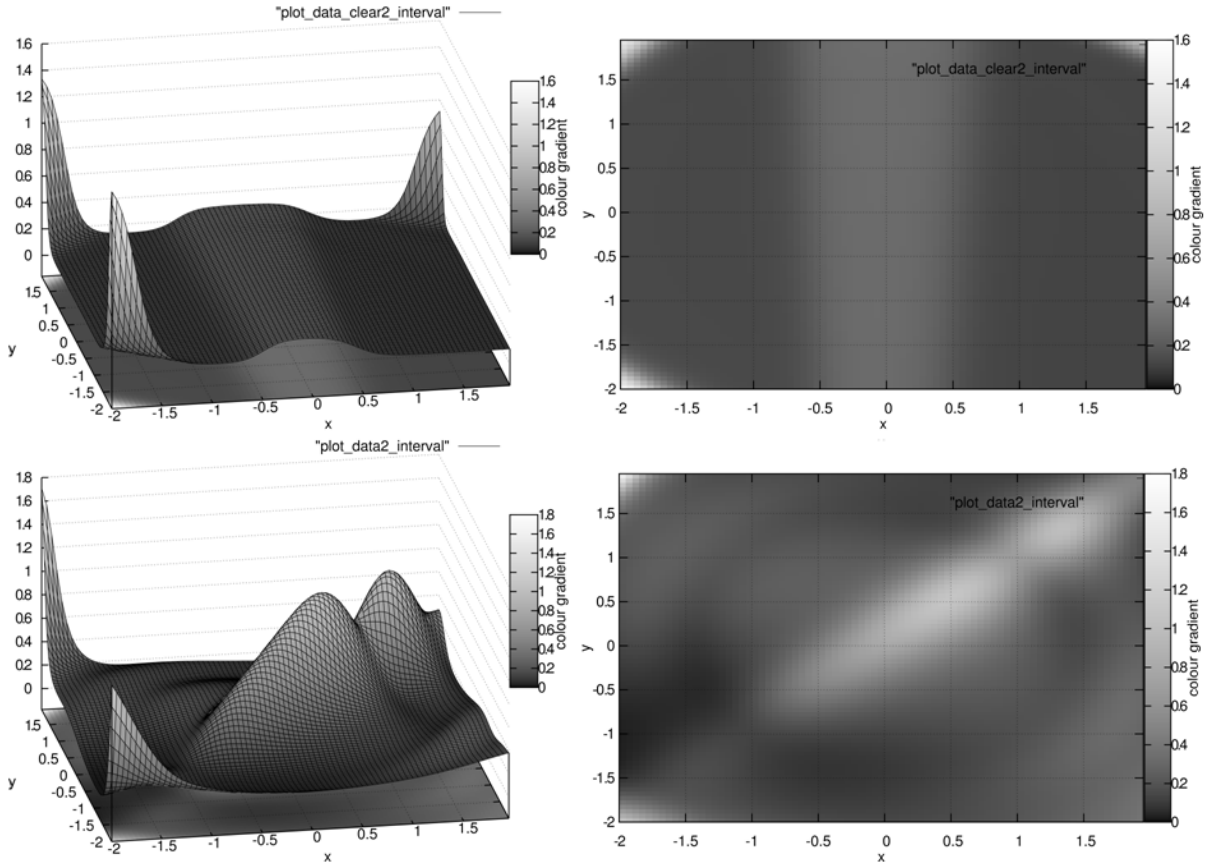

Рис. 2. Поверхность Δy после обучения НСПР без добавления шума (сверху) и с добавлением (снизу)

обучающая выборка непосредственно не используется. Следовательно, если в силу каких-то причин, например, нехватки нейронов, любая из двух НСПР обучена не верно, то некорректными будут и интервалы;

6) для аппроксимации $G(\boldsymbol{x}, w)$ рекомендуется использовать НСПР, по сложности не уступающей НСПР для $F(\boldsymbol{x}, w)$.

В результате получен механизм обрабатывающий детерминированные данные для построения модели, учитывающий неопределённость изучаемого объекта. Результаты вычислений в дальнейшем могут использоваться в широком круге алгоритмов принятия решений, системах нечёткого управления, самонастраивающихся систем.

СПИСОК ЛИТЕРАТУРЫ

1. *Хайкин С.* Нейронные сети. – М. : Вильямс, 2006. – 1104 с.

2. *Добронец Б. С.* Интервальные числа // Интервальная математика: Учеб. пособие. – Красноярск, 2004. – С. 31–33.

3. *Пархоменко С. С., Леденёва Т. М.* Обучение нейронных сетей методом Левенберга-Марквардта в условия большого количества данных // Вестник Воронежского ун-та. Серия: Системный анализ и информационные технологии. – 2014. – №2. – С. 98–106.

4. Ведение в теорию нейронных сетей // Нечеткая логика: алгебраические основы и приложения: Монография / С. Л. Блюмин, И. А. Шуйкова, П. В. Сараев, И. В. Черпаков. – Липецк : ЛЭГИ, 2002. – С. 85–91.

5. Скользящий контроль. URL: http://www.machinelearning.ru/wiki/index.php?oldid=12635 (дата обращения 18.03.2015).

6. *Сараев П. В.* Суперпозиционное линейно-нелинейное нейроструктурное моделирование: Дисс… кандидата наук / Павел Викторович Сараев; Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Липецкий государственный технический университет». – 2012.

7. *MacKay D.* Bayesian Methods for Adaptive Models: PhD thesis. – California Institute of Technology, 1991. – 98 с.

8. *Нужный А. С., Шумский С. А.* Регуляризация Байеса в задаче аппроксимации функции многих переменных // Математическое моделирование. – 2003. – Т. 15, № 9. – С. 55–63.

9. *Foresee F. D., Hagan M. T.* Gauss-Newton approximation to Bayesian learning // Proceedings of the 1997 international joint conference on neural networks. – Piscataway: IEEE, 1997. – Т. 3. – С. 1930 – 1935.

10. *Poland J.* On the Robustness of Update Strategies for the Bayesian Hyperparameter α. – 2001. – Т. 10. – С. 2.

11. *Souza C.* Neural network learning by the levenberg-marquardt algorithm with Bayesian regularization (part 2). – 2009. – URL: http://crsouza-blog.azurewebsites.net/2009/11/neural-network-learning-by-the-levenberg-marquardt-algorithm-with-bayesian-regularization-part-2/ (дата обращения 13.03.2015)

12. *Сигель Э.* Интервалы предсказания // Практическая бизнес-информатика : Пер. с англ. – М. : Издательский дом «Вильямс», 2002. – С. 423.

13. Comprehensive Review of Neural Network-based Prediction Intervals and New Advances // Neural Networks, IEEE Transactions on vol. 22, №9 / Abbas Khosravi, Saeid Nahavandi, Doug Creighton, Amir F Atiya. – IEEE, 2011. – P. 1341–1356.

14. *Zapranis A., Livanis E.* Prediction intervals for neural network models // Proceedings of the 9th WSEAS International Conference on Computers. – World Scientific and Engineering Academy and Society (WSEAS), 2005. – С. 76.

Пархоменко Станислав Сергеевич – аспирант кафедры вычислительной математики и прикладных информационных технологий Воронежского государственного университета.
E-mail: ibm8086@yandex.ru

Леденёва Татьяна Михайловна – д. т. н., проф., заведующий кафедрой вычислительной математики и прикладных информационных технологий Воронежского государственного университета.
Тел.: +7(473) 228-11-60 (доб. 1470)
E-mail: ledeneva-tm@yandex.ru

Parkhomenko Stanislav Sergeevich – postgraduate student of the Department of Computational Mathematics and Applied Information Technology, Voronezh State University.
E-mail: ibm8086@yandex.ru

Ledeneva Tatyna Michaylovna – Doctor of Technic Sciences, Professor, Head of the Department of Computational Mathematics and Applied Information Technology, Voronezh State University.
Tel.: +7(473) 228-11-60 (1470).
E-mail: ledeneva-tm@yandex.ru