О построении явной архитектуры нейронной сети, приближающей кусочно-линейные функции

В. Г. Шишляков[1]

В работе рассматривается вопрос о нахождении оценки сверху параметров архитектуры нейронной сети, которая хорошо приближает зависимости, описывающиеся кусочно-линейными функциями. Основной результат работы заключен в теореме, утверждающей, что любую наперед заданную кусочно-линейную функцию можно приблизить сколь угодно точно нейронной сетью с активационными функциями сигмоидного типа на достаточно объемном множестве. Доказательство данной теоремы конструктивно, то есть в ней строится архитектура нейронной сети, удовлетворяющая вышеописанным свойствам.

Ключевые слова: схемы функциональных элементов, нейронные сети, архитектура, аппроксимация функций, оценка сверху, кусочно-линейные функции.

1. Введение

Истоки подобных задач восходят еще к 1900 году, когда Д. Гильберт сформулировал список существенных проблем математики, в котором под номером 13 был вопрос о представимости функции n переменных в виде суперпозиции функций меньшего числа переменных.

В пятидесятых годах XIX века А.Н. Колмогоров [1], [2], [3] и В.И. Арнольд [4], [5] показали, что любую непрерывную функцию n переменных можно представить в виде суперпозиции одноместных функций и операции сложения.

В дальнейшем про представления, описанные в [1], [2], [3], [4], [5] вспомнили во время развития искусственных нейронных сетей. Однако данные представления были доказаны в неконструктивной форме – для каждой непрерывной функции n переменных требовалось искать новые одноместные функции, участвующие в суперпозиции. При этом явного алгоритма поиска таких функций представлено не было. Поэтому данные представления не нашли применения в области нейронных сетей, но

[1] *Шишляков Владимир Геннадьевич* — аспирант каф. общих проблем управления мех.-мат. ф-та МГУ, e-mail: bolotmaks@yandex.ru.

Shishlyakov Vladimir Gennad'evich — graduate student, Lomonosov Moscow State University, Faculty of Mechanics and Mathematics, Chair of General Problems of Control.

направили исследователей в нужное направление – обоснование способностей нейронных сетей восстанавливать или приближать определенные классы функций.

Исследования в области восстановления и аппроксимации функций нейронными сетями начались с работ У. Мак-Каллока и У. Питтса [6], [7], в которых давалось описание математической модели нейрона, а также было доказано, что булевские функции и конечные автоматы могут быть представлены нейронными сетями.

Однако на тот момент было не ясно, как производить обучение нейронных сетей, то есть как производить настройку их синаптических весов. Первый алгоритм обучения нейронных сетей был разработан Ф. Розенблаттом [8], [9]. Им была разработана модель нейронной сети, названная им перцептроном, и описан алгоритм обучения такой модели.

Было показано, что некоторые задачи могут быть решены перцептронами Розенблатта эффективнее, чем компьютерами с классической архитектурой. Однако в дальнейшем М. Минский и С. Пейперт [10] выяснили, что область применимости перцептронов имеет серьезные ограничения. В частности, ими было показано, что для некоторых задач, которые могут быть решены перцептроном Розенблатта, может потребоваться либо очень большое число нейронов, либо очень большое количество времени.

Ограничения [10] были сняты при замене функций активации нейронов с пороговых на сигмоидные. В конце восьмидесятых Г. Цыбенко [11], К. Фунахаши [12] и К. Хорник [13] независимо показали, что любую непрерывную на компакте $K \subset \mathbb{R}^n$ функцию f можно аппроксимировать в равномерной метрике многослойной нейронной сетью с линейными функциями активации в последнем слое и функциями сигмоидного типа во внутренних слоях. При этом в работах [11] и [13] были получены результаты не только для непрерывных функций, но и для функций пространства L_1.

Вместе с исследованиями [14], [15], [16], в которых был открыт и развит алгоритм обратного распространения ошибки, работы [11], [12], [13] дали теоретическое обоснование разумности использования нейронных сетей с сигмоидальными функциями активации и их автоматического обучения при помощи алгоритма обратного распространения ошибки.

Однако доказательства в работах [11], [12], [13] были неконструктивны, при этом не уточнялось, какое количество нейронов требуется взять в каждом слое имеющейся нейронной сети, чтобы приблизить заранее выбранную непрерывную функцию. Таким образом, вопрос о выборе разумной архитектуры нейронной сети оставался исследованным не до конца, так как в работах указывалось достаточное число слоев в нейронных сетях, но не количество нейронов в каждом слое.

Кроме того, в данных работах аппроксимации осуществлялись ступенчатыми функциями, которые позволяют с легкостью решать задачи классификации и управления, так как решения подобных задач основываются на аппроксимации кусочно-постоянных функций, но бывают крайне неудобными, например, в задачах регрессии. Так, к примеру, для приближения линейной функции на компакте при помощи ступенчатой функций с приемлемым качеством потребуется тем больше нейронов, чем точнее требуется приближение. К тому же, даже при малейшем выходе точки пространства входных данных за компакт K, качество приближения функции нейронной сетью, построенной подобным образом, может стремительно падать. Это бывает неудобно, если предполагается, что входные данные, на которых будет использоваться нейронная сеть, не имеют каких-либо ограничений.

В 2003 году в работе Д.В. Алексеева [17] было показано, что в интегральной метрике с весом Чебышева-Эрмита возможно приближение произвольной измеримой по Лебегу функции n переменных двухслойной нейронной сетью, причем функции активации первого слоя могут быть заданы заранее, а второго – линейны. Однако в работе [17] не указывалась какая-либо оценка числа нейронов в каждом слое нейронной сети, а интегральная метрика с весом Чебышева-Эрмита являлась более слабой по сравнению с равномерной метрикой.

Наконец, в 2009 году в работе В.С. Половникова [18], нейронные сети, построенные из нейронов модели Мак-Каллока и Питтса, были рассмотрены с точки зрения схем функциональных элементов [19]. Также в работе [18] было доказано, что любую кусочно-линейную функцию (не обязательно непрерывную) можно представить в виде схемы функциональных элементов над базисом $B_1 = \{c, \gamma \cdot x, \sum_n(x_1, ..., x_n), \theta(x), F(x, y)\}$, где $F(x,y) = \begin{cases} x, & y \geqslant 0 \\ 0, & y < 0 \end{cases}$, $\theta(x) = \begin{cases} 1, & x \geqslant 0 \\ 0, & x < 0 \end{cases}$. Доказательство в работе [18] было конструктивным, причем в доказательстве были даны оценки количества функциональных элементов, достаточных для восстановления любой кусочно-линейной функции.

Однако базис B_1 не подходил для обучения нейронных сетей градиентными методами из-за того, что в нем имелись функции со всюду нулевыми производными.

Изначальная цель данной работы - решить описанную проблему. Но в дальнейшем цель была расширена до исследования более обобщенного базиса $B_2 = \{c, \gamma \cdot x, \sum_n(x_1, ..., x_n), \prod_n(x_1, ..., x_n), \psi(x)\}$, где $\psi(x)$ - это некоторая произвольная функция сигмоидного типа [17].

Базис B_2 был выбран исходя из следующих умозаключений. Если возможно восстановление произвольной кусочно-линейной функции схемами функциональных элементов над базисом B_1, то, в силу того, что

$F(x,y) = x \cdot \theta(y)$, такое восстановление возможно и схемами функциональных элементов над базисом
$B_3 = \{c, \gamma \cdot x, \sum_n(x_1,...,x_n), \prod_n(x_1,...,x_n), \theta(x)\}$. При замене в базисе B_3 функции $\theta(x)$ на $\sigma(x) = \frac{1}{1+e^{-x}}$, получается базис, в котором все элементы являются дифференцируемыми функциями, и в котором можно надеяться хотя бы на приближение кусочно-линейных функций схемами функциональных элементов (так как функция $\sigma(x)$ является аппроксимацией функции $\theta(x)$). Но далее возникает интерес в рассмотрении еще более широкого класса базисов, получаемых из B_3 заменой функции $\theta(x)$ на произвольную функцию $\psi(x)$ сигмоидного типа, и исследования вопроса аппроксимации кусочно-линейных функций схемами функциональных элементов в полученных базисах.

Таким образом, базис B_2 является более близким к классическому базису Мак-Каллока и Питтса, а также более подходящим для общепринятого подхода обучения методом обратного распространения ошибки (в том случае, когда функция $\psi(x)$ является дифференцируемой).

Очевидно, что в базисе B_2 задача восстановления кусочно-линейной функции в общем виде (когда $\psi(x)$ - произвольная сигмоидная функция) невозможна. Поэтому в работе решается задача аппроксимации, а именно, доказывается, что в базисе B_2 любую кусочно-линейную функцию можно приблизить на некотором компакте нейронной сетью со сколь угодно большой точностью.

Причем доказательство конструктивно, а в построенной нейронной сети указывается оценка сверху количества нейронов в каждом слое сети. В силу того, что в данной работе вместо нелинейной сложности и глубины [18] используется другая оценка построенной нейронной схемы, потребовалось ввести несколько дополнительных определений, уточняющих понятия нейрона и слоя нейронной сети в терминах схем функциональных элементов.

Также стоит отметить, что в случае небольшого выхода точки пространства входных данных за пределы компакта, внутри которого производилось обучение нейронной сети, данная модель будет работать примерно с той же погрешностью, что и внутри компакта. Хотя погрешность нейронной сети постепенно увеличивается при удалении точки пространства входных данных от компакта, на котором оценивалась точность сети.

Таким образом, основной результат данной работы является конструктивным аналогом теоремы Цыбенко [11], но только для нейронных сетей, построенных над видоизмененным базисом, который удобен как для обучения нейронных сетей классическими градиентными методами (при выборе, например, $\psi(x) = \sigma(x) = \frac{1}{1+e^{-x}}$), так и для решения задач регрессии при помощи нейронных сетей. При этом в теореме, рассмот-

ренной в данной работе, дается оценка количества нейронов на каждом слое, при котором можно подобрать веса нейронной сети так, чтобы она приближала выбранную кусочно-линейную функцию с заданной точностью.

2. Основные понятия и формулировка результата

Для начала определим основные понятия, которые используются в данной статье. Нейронные сети можно рассматривать с двух точек зрения. С одной стороны на них можно смотреть, как на функции с большим количеством подбираемых параметров (весов), а с другой стороны – как на схемы, реализующие эти функции (тоже с большим количеством подбираемых параметров). Поэтому вполне логично при рассмотрении искусственных нейронных сетей со схематической точки зрения ввести понятие базиса нейронной сети. Помимо этого, следуя ссылкам [18], [19] и [20], напомним определения и обозначения основных объектов нейронных сетей.

Определение 1. Базисом будем называть некоторый набор функциональных элементов, где каждый функциональный элемент представляет из себя пару $(S, f(x_1, ..., x_n))$, в которой $f(x_1, .., x_n) : \mathbb{R}^n \longrightarrow \mathbb{R}$, а S - сопоставленный ей графический объект с n входными стрелками и одной выходной (кратко – входы и выход объекта S). Входам объекта S приписаны слева направо переменные $x_1, ..., x_n$ функции f, выходу приписан выход функции f.

Стоит отметить, что базис нейронной сети не является базисом с обычной математической точки зрения, так как он часто является избыточным (то есть существуют элементы базиса, выражаемые через другие элементы базиса).

В классической модели Мак-Каллока Питтса [6] принимается базис (1), приведенный ниже.

$$B_1 = \{c, \gamma \cdot x, \sum_n (x_1, ..., x_n), \theta(x)\} \tag{1}$$

В базисе (1) используются следующие классы функций:

1) Сумматор - каждая функция данного класса суммирует определенное количество входных аргументов и обозначается $\sum_n (x_1, ..., x_n)$.

2) Константа - каждая функция данного класса выдает константу (у данной функции нет входных аргументов, каждый раз, когда на схему приходят входные сигналы, константа выдает одинаковое заранее определенное значение).

3) Усилитель (умножение на константу) - в данном классе каждая функция умножает пришедший на вход аргумент x на фиксированную константу γ.

4) Функция активации - в модели Мак-Каллока Питтса эта функция единственна и выглядит так
$\theta(x) = \begin{cases} 1, & x \geqslant 0 \\ 0, & x < 0 \end{cases}$.

Графические изображения данных элементов приведены на рис. 1 (а, б, в, г).

Определение 2. *Функцию* $\psi(x) : \mathbb{R} \longrightarrow \mathbb{R}$ *будем называть сигмоидной [13], [17], если она не убывает на* $\mathbb{R}$ *и выполняется, что* $\lim_{x \longrightarrow -\infty} \psi(x) = 0$, $\lim_{x \longrightarrow +\infty} \psi(x) = 1$.

Рассмотрим следующий базис:

$$B_2 = \{c, \gamma \cdot x, \sum_n (x_1, ..., x_n), \prod_n (x_1, ..., x_n), \psi(x)\} \tag{2}$$

Базис (2) отличается от классического базиса (1) тем, что в нем добавлено семейство функций $\prod_n (x_1, ..., x_n)$, которое определяется аналогично семейству $\sum_n (x_1, ..., x_n)$, а функция активации $\theta(x)$ заменена на $\psi(x)$ - произвольную сигмоидную функцию.

Функции $\prod_n (x_1, ..., x_n)$ будем обозначать на схемах, как на рис. 1 (д), а $\psi(x)$ - как на рис. 1 (е).

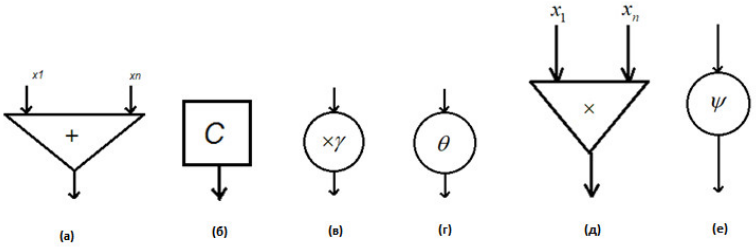

Рис. 1. Функциональные элементы рассматриваемых базисов

Подробнее о том, как строятся схемы функциональных элементов из элементов любого из рассматриваемых в данной работе базисов, описано в [18].

Определение 3. *Нейроном в базисе* (2) *будем называть всякую схему, вычисляющую одну из следующих функций*

$$\varphi(\sum_{i=1}^{n}(w_i \cdot x_i) + c) \tag{3}$$

или

$$\varphi(\prod_{i=1}^{n}(w_i \cdot x_i) + c) \tag{4}$$

В формулах (3) *и* (4) *функция* $\varphi(x)$ *называется активационной функцией. В качестве* $\varphi(x)$ *может быть выбрана либо* $\psi(x)$*, либо* x*.*

В дальнейшем, для краткости, нейроны в схемах будем обозначать, как на рис. 2 (а, б). Если же функция активации φ является тождественной, то такие нейроны будем обозначать, как на рис. 2 (в, г).

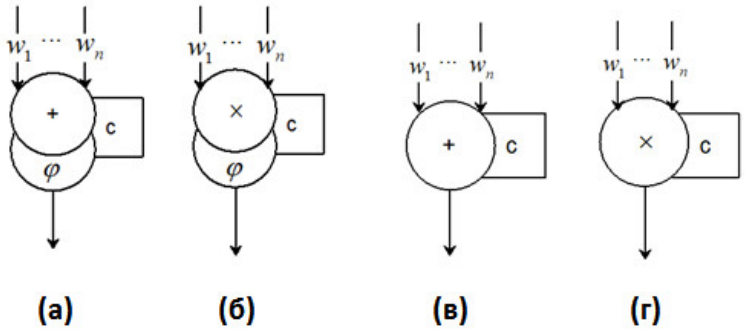

Рис. 2. Графические обозначения нейронов (3) и (4)

Следует отметить, что самой распространенной [20] схемой нейронов (3) и (4) является схема, изображенная на рис. 3. В дальнейших выкладках именно такие схемы будут заменяться на обозначения нейронов.

Определение 4. *Все нейроны вида* (3) *будем называть нейронами-сумматорами, а нейроны вида* (4) *- нейронами-продукторами.*

Определение 5. *Введем понятие слоя нейронной сети.*

1) *Множество нейронов, все входы которых не подсоединены ни к каким выходам каких-либо функциональных элементов, назовем нейронами первого слоя.*

2) *Пусть определено множество нейронов* n*-го слоя. Тогда* $n+1$*-ым слоем назовем все нейроны, для которых выполняются одновременно следующие условия:*

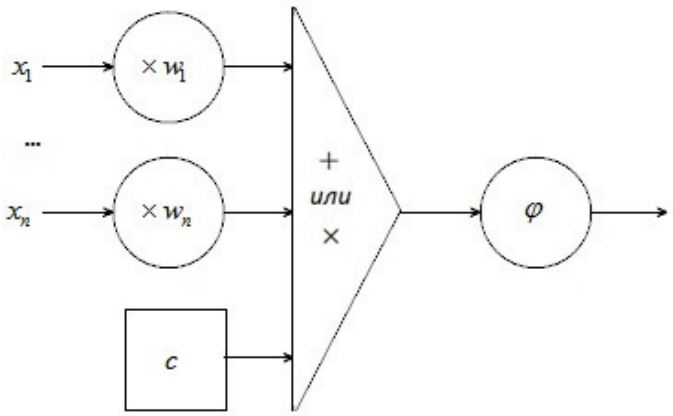

Рис. 3. Схема нейрона

а) Хотя бы один вход подсоединен к выходу нейрона n-го слоя

б) Все оставшиеся входы подсоединены либо к выходам нейронов из слоев $\{1, 2, ..., n\}$*, либо не подсоединены ни к каким нейронам (тогда считается, что на вход принимаются входные данные).*

Таким образом, можно комбинировать не отдельные элементы базиса схемы, а целые схемы, реализующие нейроны. При комбинации нейронов друг с другом, будут получаться различные функции, которые и будут исследоваться в данной работе. Особый интерес для данного исследования представляют так называемые кусочно-линейные функции. Дадим их определение, следуя определениям из [18].

Определение 6. *Пусть* $\overline{x} = (x_1, ..., x_n)$ *(входные сигналы),* $l_1, ..., l_k$ *- некоторые гиперплоскости, определяемые выражениями* $l_i = \{x \in \mathbb{R}^n | \langle \overline{a_i}, \overline{x} \rangle + c_i = 0\}$ *(здесь* $\langle \overline{a_i}, \overline{x} \rangle = \sum_{j=1}^{n} a_{ij} \cdot x_j$ *- скалярное произведение векторов,* c_i *константа,* $\overline{a_i} \neq \overline{0}$*). Также обозначим* $l_i(\overline{x}) = \langle \overline{a_i}, \overline{x} \rangle + c_i$*.*

Отметим, что все пространство $\mathbb{R}^n$ *разбивается этими гиперплоскостями* $l_1, ..., l_k$ *на классы эквивалентности. Вектор-функция* $\sigma(\overline{x}) = (sgn(\overline{a_1} \cdot \overline{x} + c_1), ..., sgn(\overline{a_k} \cdot \overline{x} + c_k))$ *называется сигнатурой вектора* $\overline{x}$ *[18]. Каждая ее компонента показывает расположение точки* $\overline{x}$ *относительно соответствующей ей гиперплоскости из набора* $\{l_1, ..., l_k\}$*. Поэтому данная функция однозначно определяет, в каком куске пространства лежит точка* $\overline{x} = (x_1, ..., x_n) \in \mathbb{R}^n$*.*

Сигнатурой класса будем называть сигнатуру любого вектора из этого класса. Определение корректно, так как у всех точек одного класса одинаковые сигнатуры [18].

Определение 7. *Пусть пространство* $\mathbb{R}^n$ *разбивается на классы* $R^1, ..., R^s$ *гиперплоскостями* $l_1, ..., l_k$. *Будем говорить, что* $f(\overline{x}) \in PL$ *(является кусочно-линейной), если* $f(\overline{x})|_{R^j} = \overline{b_j} \cdot \overline{x} + d_j$ *(то есть сужение на каждый из классов* $R^1, ..., R^s$ *является линейной функцией,* $\overline{b_j}, d_j = const$*).*

Также введем несколько полезных обозначений. Пусть $l_1, ..., l_k$ - гиперплоскости. Возьмем $\forall \xi > 0$ и рассмотрим множества

$L_{i,\xi} = \{\overline{x} \in \mathbb{R}^n || l_i(\overline{x})| < \xi\}, i = 1, ..., k$. Обозначим $L_\xi = \bigcup\limits_{i=1}^{k} L_{i,\xi}$.

Выражением $O_R(\overline{x})$ будем обозначать окрестность радиуса R некоторой точки $\overline{x}$.

Основным результатом данной статьи является теорема, в которой утверждается, что любую кусочно-линейную функцию, заданную относительно гиперплоскостей $l_1, ..., l_k$, можно приблизить нейронной схемой над базисом (2) со сколь угодно большой точностью всюду вне множества L_ξ, где ξ может быть сколь угодно близким к нулю числом.

3. Основные результаты

Теорема 1 (о приближении кусочно-линейной функции)**.** *Пусть* $l_1, ..., l_k$ *- гиперплоскости, которые разбивают пространство* $\mathbb{R}^n$ *на* s *классов эквивалентности* $R^1, ..., R^s$*, из которых* s' *классов обладают такими сигнатурами* $(sgn(\overline{a_1} \cdot \overline{x} + c_1), ..., sgn(\overline{a_k} \cdot \overline{x} + c_k))$*, что* $sgn(\overline{a_i} \cdot \overline{x} + c_i) \neq 0, i = 1, ..., k$*, а* $f(\overline{x})$ *- кусочно-линейная функция, заданная над данными классами эквивалентности.*

Тогда $\forall \varepsilon > 0, \forall \xi > 0, \forall R > 0$ *существует нейронная сеть* $G(\overline{x})$ *над базисом* (2) *такая, что выполняется* $\sup\limits_{\overline{x} \in O_R(\overline{0}) \setminus L_\xi} |G(\overline{x}) - f(\overline{x})| < \varepsilon$*. Причем данная нейронная сеть обладает следующей архитектурой:*

1. На первом слое потребуется не более $2k$ *нейронов-сумматоров, имеющих функцию активации* $\varphi(x) = \psi(x)$*;*

2. На втором слое потребуется $2s'' \leqslant 2s'$ *нейронов, из которых* s'' *нейронов имеют функцию активации* $\varphi(x) = \psi(x)$*, а остальные* s'' *нейронов –* $\varphi(x) = x$*. При этом каждый нейрон с функциями активации* $\varphi(x) = x$ *на данном слое принимает на вход кроме выходов нейронов предыдущего слоя вектор* $\overline{x} = (x_1, ..., x_n)$*, который является копией вектора, подающегося на нейроны входного слоя сети;*

3. На третьем слое потребуется не более s' нейронов-продукторов с тождественной функцией активации;

4. На четвертом слое потребуется один нейрон-сумматор с тождественной функцией активации.

Доказательство. Зафиксируем произвольные $\xi > 0, R > 0$ и рассмотрим множество $O = O_R(\overline{0}) \setminus L_\xi$. Покажем, что $\exists G(\overline{x})$, реализуемая нейронной сетью над базисом (2), такая что $\sup\limits_{\overline{x} \in O} |G(\overline{x}) - f(\overline{x})| < \varepsilon$.

Зафиксируем произвольное $\varepsilon > 0$. Константу d положим такой, чтобы $\psi(x + d) > \frac{1}{2}$ при $x > 0$ и $\psi(x + d) \leqslant \frac{1}{2}$ при $x \leqslant 0$. Такая константа существует в силу того, что $\lim\limits_{x \to -\infty} \psi(x) = 0$, $\lim\limits_{x \to +\infty} \psi(x) = 1$ и ψ является неубывающей функцией. Положим $c > 1$ таким, чтобы $|\psi(c \cdot \xi + d) - \frac{1}{2}| \geq \varepsilon'$ и $|\psi(c \cdot (-\xi) + d) - \frac{1}{2}| \geq \varepsilon'$, где $\varepsilon' = \frac{1}{2} \cdot \frac{k}{(k+1)}$. Это всегда можно сделать, так как по условию $\lim\limits_{x \to -\infty} \psi(x) = 0$ и $\lim\limits_{x \to +\infty} \psi(x) = 1$.

Но тогда выполняется:

$$\psi(c \cdot l_i(\overline{x}) + d) \in \begin{cases} [\frac{1}{2} + \varepsilon', 1], & l_i(\overline{x}) \geqslant \xi \\ [0, \frac{1}{2} - \varepsilon'], & l_i(\overline{x}) \leqslant -\xi \end{cases} \tag{5}$$

$$\psi(c \cdot (-l_i(\overline{x})) + d) \in \begin{cases} [\frac{1}{2} + \varepsilon', 1], & l_i(\overline{x}) \leqslant -\xi \\ [0, \frac{1}{2} - \varepsilon'], & l_i(\overline{x}) \geqslant \xi \end{cases} \tag{6}$$

Рассмотрим теперь класс R^j, обладающий сигнатурой $(\sigma_1^j, ..., \sigma_k^j)$, где все $\sigma_i^j \neq 0$. Построим функцию $\sum\limits_{i=1}^{k} \pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x})$, где $\pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) = \begin{cases} \psi(c \cdot l_i(\overline{x}) + d), & \sigma_i^j = 1 \\ \psi(c \cdot (-l_i(\overline{x})) + d), & \sigma_i^j = -1 \end{cases} = \psi(\sigma_i^j \cdot c \cdot l_i(\overline{x}) + d)$.

Из (5) и (6) следует, что $\forall \overline{x} \in O$ выполняется:

$$\pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) \in [\frac{1}{2} + \varepsilon', 1], \text{если } sgn(l_i(\overline{x})) = \sigma_i^j \tag{7}$$

$$\pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) \in [0, \frac{1}{2} - \varepsilon'], \text{если } sgn(l_i(\overline{x})) \neq \sigma_i^j \tag{8}$$

Но тогда получаем, что $\forall \overline{x} \in O$, если $\overline{x} \in R^j$, то $(sgn(l_1(\overline{x})), ..., sgn(l_k(\overline{x}))) = (\sigma_1^j, ..., \sigma_k^j)$ и тогда из (7) получаем, что $\pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) \geqslant \frac{1}{2} + \varepsilon'$, $i \in \{1, ..., k\}$. А тогда $\sum\limits_{i=1}^{k} \pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) \geqslant \sum\limits_{i=1}^{k} (\frac{1}{2} + \varepsilon') = \frac{1}{2}k + \varepsilon' k$.

Если же $\forall \overline{x} \in O$, но $\overline{x} \notin R^j$, то вектор $(sgn(l_1(\overline{x}), ..., sgn(l_k(\overline{x})))$ отличается от вектора $(\sigma_1^j, ..., \sigma_k^j)$ хотя бы в одной компоненте. Пусть это компонента p, то есть $sgn(l_p(\overline{x})) \neq \sigma_p^j$. Тогда из (7) и (8) получаем, что $\pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) \leqslant 1,\ i \in \{1, ..., k\} \setminus \{p\}$ и $\pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) \leqslant \frac{1}{2} - \varepsilon',\ i = p$. Откуда $\sum\limits_{i=1}^{k} \pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) \leqslant \sum\limits_{i=1, i\neq p}^{k} \pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) + \pi_{sgn(l_p(\overline{x}))=\sigma_p^j}(\overline{x}) \leqslant k - 1 + \frac{1}{2} - \varepsilon' = k - \frac{1}{2} - \varepsilon'$.

Найдем условие на ε', при котором

$$\frac{1}{2}k + k\varepsilon' > k - \frac{1}{2} - \varepsilon' \tag{9}$$

$\frac{1}{2}k + k\varepsilon' > k - \frac{1}{2} - \varepsilon' \Leftrightarrow (k+1)\ \varepsilon' > \frac{1}{2}(k-1) \Leftrightarrow \varepsilon' > \frac{1}{2}\frac{(k-1)}{(k+1)}$. Но для взятого в начале доказательства ε' выполняется следующее неравенство:

$$\varepsilon' = \frac{1}{2}\frac{k}{(k+1)} > \frac{1}{2}\frac{(k-1)}{(k+1)}, \tag{10}$$

поэтому условие (9) выполняется.

Таким образом, взяв $\pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) = \psi(\sigma_i^j \cdot c \cdot l_i(\overline{x}) + d)$, получим, что $\forall \overline{x} \in O$ верно, что при $\overline{x} \in R^j$ сумма $\sum\limits_{i=1}^{k} \pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x})$ всегда больше той же суммы при $\overline{x} \notin R^j$.

Положим теперь $M = \frac{1}{2} \cdot ((\frac{1}{2}k + k\varepsilon') + (k - \frac{1}{2} - \varepsilon'))$. В силу (10) для выбранного ε' выполняется условие (9), из которого следует, что $\frac{1}{2}k + \varepsilon' k \neq k - \frac{1}{2} - \varepsilon'$. Поэтому $k - \frac{1}{2} - \varepsilon' < M < \frac{1}{2}k + \varepsilon' k$.

Рассмотрим функцию $\sum\limits_{i=1}^{k} \pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) - M$ и $\forall \overline{x} \in O$. Очевидно, что если $\overline{x} \in R^j$, то $\sum\limits_{i=1}^{k} \pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) - M > 0$, а если $\overline{x} \notin R^j$, то $\sum\limits_{i=1}^{k} \pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) - M < 0$.

Далее, взяв $m = min\left\{\frac{1}{2}k + k\,\varepsilon' - M, M - \left(k - \frac{1}{2} - \varepsilon'\right)\right\}$, получаем, что для $\forall \overline{x} \in O$ выполняется, что $\left|\sum\limits_{i=1}^{k} \pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) - M\right| \geqslant m$.

Теперь возьмем сколь угодно большое число $\mu > 0$ и $\forall \overline{x} \in O$. В силу того, что $\left|\sum\limits_{i=1}^{k} \pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) - M\right| \geqslant m$, получаем, что

$\frac{\mu}{m}\left(\sum\limits_{i=1}^{k} \pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) - M\right) \geqslant \mu$ при $\overline{x} \in R^j$ и

$\frac{\mu}{m}\left(\sum\limits_{i=1}^{k} \pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) - M\right) \leqslant -\mu$ при $\overline{x} \notin R^j$.

Обозначим $s(\xi) = \#\{R^j | R^j \cap O \neq \varnothing\}$. Очевидно, что $s(\xi) \in \mathbb{N} \cup \{0\}$: $s(\xi) \leqslant s' \leqslant s$, причем, функция $s(\xi)$ является не возрастающей. Другими словами, при уменьшении величины $\xi > 0$ соответствующее ей значение $s(\xi)$ не убывает.

Если $s(\xi) = 0$, то взятое $\xi > 0$ оказалось слишком большим и $O = \varnothing$. Поэтому любая функция подойдет в качестве $G(\overline{x})$, в том числе и реализуемая нейронной сетью с требуемой архитектурой и произвольными значениями ее параметров.

Поэтому здесь и далее будем полагать, что $s(\xi) \geqslant 1$, а, следовательно, $O \neq \varnothing$. Также, без ограничения общности, будем считать, что $\{R^j | R^j \cap O \neq \varnothing\} = \{1, ..., s(\xi)\}$.

Положим $\tau(\varepsilon, R, \xi) = \frac{\varepsilon}{s(\xi) \cdot \max\limits_{\overline{x} \in O} |f(\overline{x})| + 1} < \infty$. Обозначим

$\Psi_j(\overline{x}) = \psi\left(\frac{\mu}{m}\left(\sum\limits_{i=1}^{k} \pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) - M\right)\right)$, $j \in \{1, ..., s(\xi)\}$. После чего возьмем μ таким, чтобы для $\forall \overline{x} \in O$ и $\forall j \in \{1, ..., s(\xi)\}$ были выполнены следующие условия:

$$|\Psi_j(\overline{x}) - 1| < \tau(\varepsilon, R, \xi) \text{ при } \overline{x} \in R^j \tag{11}$$

и

$$|\Psi_j(\overline{x}) - 0| < \tau(\varepsilon, R, \xi) \text{ при } \overline{x} \notin R^j. \tag{12}$$

Далее рассмотрим функцию

$$G(\overline{x}) = \sum_{j=1}^{s(\xi)} (\overline{b_j} \cdot \overline{x} + d_j) \cdot \Psi_j(\overline{x}) \tag{13}$$

Возьмем $\forall \overline{x} \in O$. Очевидно, что $\exists p \in \{1, ..., s(\xi)\} : \overline{x} \in R^p$. Тогда верны следующие рассуждения:

$$|G(\overline{x}) - f(\overline{x})| = |\sum_{j=1}^{s(\xi)}(\overline{b_j}\cdot\overline{x}+d_j)\cdot\Psi_j(\overline{x}) - f(\overline{x})|_{R^p}| =$$

$$\left|\sum_{\substack{j=1,\\ j\neq p}}^{s(\xi)}(\overline{b_j}\cdot\overline{x}+d_j)\cdot\Psi_j(\overline{x}) + (\overline{b_p}\cdot\overline{x}+d_p)\cdot(\Psi_p(\overline{x})-1)\right| <$$

$$\left|\sum_{\substack{j=1,\\ j\neq p}}^{s(\xi)}\max_{\overline{x}\in O}|f(\overline{x})|\cdot\tau(\varepsilon,R,\xi)\right| + \left|\max_{\overline{x}\in O}|f(\overline{x})|\cdot\tau(\varepsilon,R,\xi)\right| =$$

$$\tau(\varepsilon,R,\xi)\cdot\sum_{j=1}^{s(\xi)}\max_{\overline{x}\in O}|f(\overline{x})| = \tau(\varepsilon,R,\xi)\cdot s(\xi)\cdot\max_{\overline{x}\in O}|f(\overline{x})| <$$

$$\tau(\varepsilon,R,\xi)\cdot(s(\xi)\cdot\max_{\overline{x}\in O}|f(\overline{x})|+1) = \varepsilon \quad (14)$$

Из (14) немедленно следует, что $\sup\limits_{\overline{x}\in O}|G(\overline{x}) - f(\overline{x})| < \varepsilon$.

Учитывая, что $\pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) = \psi(\sigma_i^j\cdot c\cdot l_i(\overline{x})+d)$, формулу (13) для $G(\overline{x})$ можно переписать в следующем виде:

$$\sum_{j=1}^{s(\xi)}(\overline{b_j}\cdot\overline{x}+d_j)\cdot\psi\left(\frac{\mu}{m}\left(\sum_{i=1}^{k}\pi_{sgn(l_i(\overline{x}))=\sigma_i^j}(\overline{x}) - M\right)\right) =$$

$$\sum_{j=1}^{s(\xi)}(\overline{b_j}\cdot\overline{x}+d_j)\cdot\psi\left(\frac{\mu}{m}\cdot\sum_{i=1}^{k}\psi(\sigma_i^j\cdot c\cdot l_i(\overline{x})+d) - \frac{\mu}{m}\cdot M\right) \quad (15)$$

Обозначив $\Delta = -\frac{\mu}{m}\cdot M = const$, $\delta_{i,j}^{+} = \begin{cases}\frac{\mu}{m}, & \text{если } \sigma_i^j = 1\\ 0, & \text{если } \sigma_i^j = -1\end{cases}$ и $\delta_{i,j}^{-} = \begin{cases}0, & \text{если } \sigma_i^j = 1\\ \frac{\mu}{m}, & \text{если } \sigma_i^j = -1\end{cases}$, в выражении (15), получаем следующее представление для функции $G(\overline{x})$:

$$G(\overline{x}) = \sum_{j=1}^{s(\xi)} (\overline{b_j} \cdot \overline{x} + d_j) \cdot \psi(\sum_{i=1}^{k} \frac{\mu}{m} \cdot \psi(\sigma_i^j \cdot l_i(\overline{x}) + d) + \Delta) =$$

$$\sum_{j=1}^{s(\xi)} (\overline{b_j} \cdot \overline{x} + d_j) \cdot \psi(\sum_{i=1}^{k} \delta_{i,j}^{+} \cdot \psi(c \cdot l_i(\overline{x}) + d) +$$

$$\sum_{i=1}^{k} \delta_{i,j}^{-} \cdot \psi(-c \cdot l_i(\overline{x}) + d) + \Delta) \quad (16)$$

Далее, делая следующие обозначения в (16):

$$c \cdot l_i(\overline{x}) + d = c \cdot (a_{i1}x_1 + ... + a_{in}x_n) =$$
$$(ca_{i1})x_1 + ... + (ca_{in})x_n + (c \cdot a_{i0} + d) = p_i(\overline{x})$$

$$-c \cdot l_i(\overline{x}) + d = -c \cdot (a_{i1}x_1 + ... + a_{in}x_n) =$$
$$(-ca_{i1})x_1 + ... + (-ca_{in})x_n + (-c \cdot a_{i0} + d) = q_i(\overline{x}),$$

получаем формулу (17):

$$G(\overline{x}) = \sum_{j=1}^{s(\xi)} (\overline{b_j} \cdot \overline{x} + d_j) \cdot \psi(\sum_{i=1}^{k} \delta_{i,j}^{+} \cdot \psi(p_i(\overline{x})) +$$

$$+ \sum_{i=1}^{k} \delta_{i,j}^{-} \cdot \psi(q_i(\overline{x})) + \Delta) \quad (17)$$

Обозначим для лаконичности $(c \cdot a_{ij}) = \alpha_{ij}^{+}, (-c \cdot a_{ij}) = \alpha_{ij}^{-}$, $i = 1, ..., k, j = 1, ..., n$, а также $c \cdot a_{i0} + d = \alpha_{i0}^{+}, (-c \cdot a_{i0} + d) = \alpha_{i0}^{-}$. Тогда верно, что:

$$p_i(\overline{x}) = \alpha_{i1}^{+}x_1 + ... + \alpha_{in}^{+}x_n + \alpha_{i0}^{+} \quad (18)$$

$$q_i(\overline{x}) = \alpha_{i1}^{-}x_1 + ... + \alpha_{in}^{-}x_n + \alpha_{i0}^{-} \quad (19)$$

Изобразим схему функциональных элементов для выражения, стоящего справа от знака равенства в (17), учитывая (18) и (19), а также заменяя группы функциональных элементов, которые можно объединить в нейроны, на обозначения этих нейронов (рис. 4).

Таким образом, имеем следующую архитектуру нейронной сети. Сеть состоит из четырех слоев (не считая входного слоя), так как если рассмотреть самые длинные пути от входа к выходу, то на каждом таком

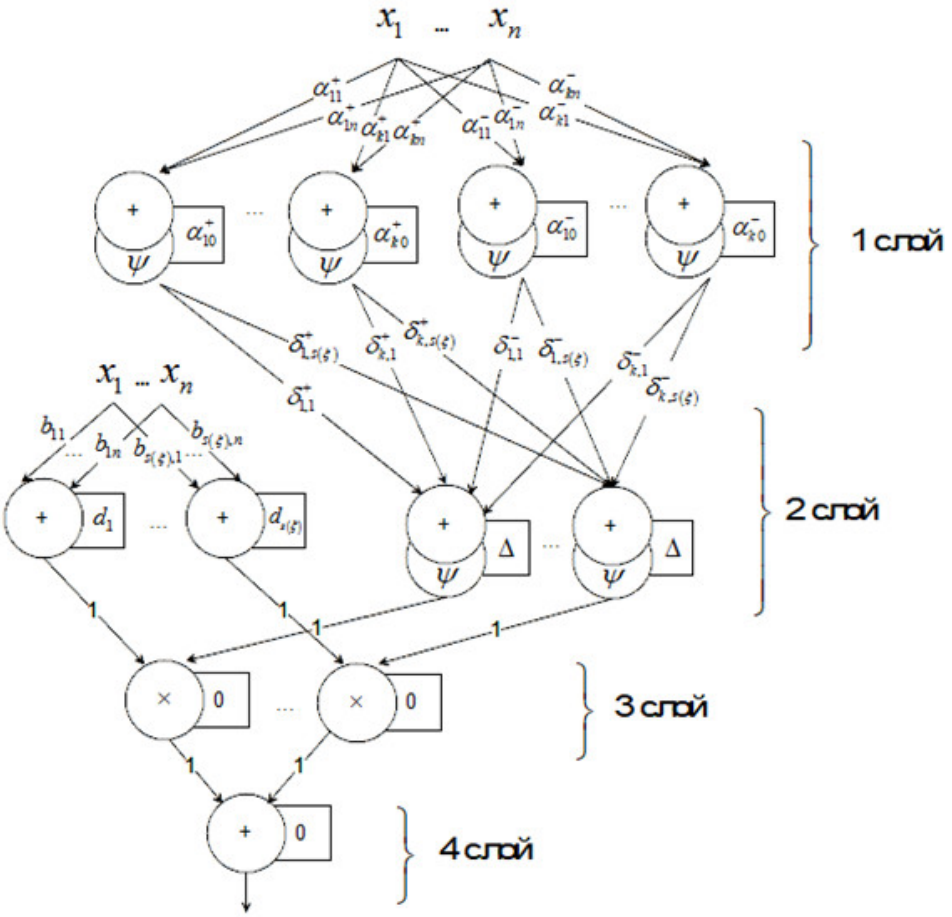

Рис. 4. Нейронная схема функции $G(\overline{x})$

пути встретится ровно 4 элемента такого вида, как на рисунках 8 и 9. Причем отметим, что:

1) На первом слое потребуется не более $2k$ нейронов-сумматоров, каждый из которых имеет функцию активации $\varphi(x) = \psi(x)$

2) На втором слое потребуется $2s(\xi) \leqslant 2s'$ нейронов-сумматоров, из которых $s(\xi)$ штук имеют функцию активации $\varphi(x) = \psi(x)$, а еще $s(\xi)$ штук – тождественную функцию активации, то есть $\varphi(x) = x$

3) На третьем слое потребуется $s(\xi) \leqslant s'$ нейронов-продукторов с тождественной функцией активации

4) На четвертом слое потребуется один нейрон-сумматор с тождественной функцией активации

Данная архитектура полностью соответствует архитектуре, заявленной в начале доказательства. □

Список литературы

[1] Колмогоров А.Н., "О представлении непрерывных функций нескольких переменных суперпозициями непрерывных функций меньшего числа переменных", *Докл. АН СССР*, **108** (1956), 2.

[2] Колмогоров А.Н., "О представлении непрерывных функций нескольких переменных в виде суперпозиции непрерывных функций одного переменного и сложения", *Докл. АН СССР*, **114** (1957), 953–956.

[3] Kolmogorov A.N., "On the Representation of Continuous Functions of Many Variables by Superposition of Continuous Functions of One Variable and Addition", *American Math. Soc. Transl*, **28** (1963), 55–63.

[4] Арнольд В.И., "О представлении любой непрерывной функции трех переменных в виде суммы функций не более двух переменных", *Докл. АН СССР*, **114**:4 (1957).

[5] Арнольд В.И., "О представлении функций нескольких переменных в виде суперпозиции функций меньшего числа переменных", *Мат. Просвещение*, **3** (1958), 41–61.

[6] McCulloch W.S., Pitts W., "A logical calculus of the ideas immanent nervous activity", *Bull. Of math. Biophysics*, **5** (1943), 115–133.

[7] Мак-Каллок У., Питтс У., *Автоматы.* V. 5: *Логическое исчисление идей, относящихся к нервной активности*, ИЛ, М., 1956.

[8] Розенблатт Ф, *Принципы нейродинамики. Перцептрон и теория механизмов мозга*: *Логическое исчисление идей, относящихся к нервной активности*, Мир, М., 1965.

[9] Rosenblat F, *Principles of Neurodynamics: Perceptrons and the Theory of Brain mechanisms*, Spartan, Washington D.C., 1962.

[10] Минский М., Пейперт С., *Перцептроны*, Мир, М., 1971.

[11] Cybenko G., "Approximations by superpositions of sigmoidal functions", *Math. Control, Signals, Systems*, **2** (1989), 303–314.

[12] Funahashi K., "On the approximate realization of continuous mappings by neural networks", *Neural Networks*, **2**:3 (1989), 183–192.

[13] Hornick K., Stinchcombe M., White H., "Multilayer feedforward networks are universal approximations", *Neural Networks*, **2**:5 (1989), 359–366.

[14] Werbos P.J., *Beyond regression: New tools for prediction and analysis in the behavioral sciences*, Ph.D. Thesis, Harvard University, Cambridge, MA, 1974.

[15] Saarinen S., R.B. Bramley and G. Cybenko, "Neural networks, backpropagation and automatic differentiation", *Automatic Differentiation of Algorithms: Theory, Implementation and Application*, eds. A. Griewank and G.F. Corliss, SIAM, Philadelphia, 1992, 31–42.

[16] Werbos P.J., *Backpropagation through time: What it does and how to do it.* V. 78, Proceedings of the IEEE, 1990, 1550–1560.

[17] Алексеев Д.В., "Приближение функций нескольких переменных нейронными сетями", *Интеллектуальные системы*, **7**:1-4 (2003), 191–205.

[18] Половников В.С., *Об оптимизации структурной реализации нейронных сетей*, дисс. ... канд. физ.-матем. наук, МГУ, Москва, 2007.

[19] Яблонский С.В., *Введение в дискретную математику*, «Наука», Москва, 1986.

[20] Haykin S., *Neural Networks. A Comprehensive Foundation*, Prentice Hall International, Inc., Canada, 1999.

On the construction of an explicit neural network architecture that approximates particle-linear functions

Shishlyakov V.G.

This work considers the question of discovering an upper-bound estimation of parameters quantity of neural network architecture well-approximating particle-linear dependances. The main result of this article consists of the theorem asserting that any particle-linear function can be approximated with any degree of precision on the big part of space by neural network with sigmoidal activation functions. This theorem has a constructive proof, i.e. neural network architecture with mentioned features building explicitly.

Keywords: schemes of functional elements, neural networks, architecture, approximation, upper-bound estimation, particle-linear functions.

References

[1] Kolmogorov A.N., "Representation of continuous functions of several variables by superpositions of continuous functions of fewer variables", *Reports of the USSR Academy of Sciences*, **108** (1956), 2 (In Russian).

[2] Kolmogorov A.N., "Representation of continuous functions of several variables as a superposition of continuous functions of one variable and addition", *Reports of the USSR Academy of Sciences*, **114** (1957), 953–956 (In Russian).

[3] Kolmogorov A.N., "On the Representation of Continuous Functions of Many Variables by Superposition of Continuous Functions of One Variable and Addition", *American Math. Soc. Transl*, **28** (1963), 55–63 (In Russian).

[4] Arnold V.I., "Representation of any continuous function of three variables as a sum of functions of at most two variables", *Reports of the USSR Academy of Sciences*, **114**:4 (1957) (In Russian).

[5] Arnold V.I., "Representation of functions of several variables as a superposition of functions of a smaller number of variables", *Math Education*, **3** (1958), 41–61 (In Russian).

[6] McCulloch W.S., Pitts W., "A logical calculus of the ideas immanent nervous activity", *Bull. Of math. Biophysics*, **5** (1943), 115–133.

[7] McCulloch I., Pitts W., *Automatic machines*. V. 5: *Logical calculus of ideas related to nervous activity*, IL, Moscow, 1956 (In Russian).

[8] Rosenblatt F., *Principles of neurodynamics. Perceptron and the theory of brain mechanisms*: *Logical calculus of ideas related to nervous activity*, MIR, Moscow, 1965 (In Russian).

[9] Rosenblat F, *Principles of Neurodynamics: Perceptrons and the Theory of Brain mechanisms*, Spartan, Washington D.C., 1962.

[10] Minsky M., Peipert S., *Perceptrons*, MIR, Moscow, 1971 (In Russian).

[11] Cybenko G., "Approximations by superpositions of sigmoidal functions", *Math. Control, Signals, Systems*, **2** (1989), 303–314.

[12] Funahashi K., "On the approximate realization of continuous mappings by neural networks", *Neural Networks*, **2**:3 (1989), 183–192.

[13] Hornick K., Stinchcombe M., White H., "Multilayer feedforward networks are universal approximations", *Neural Networks*, **2**:5 (1989), 359–366.

[14] Werbos P.J., *Beyond regression: New tools for prediction and analysis in the behavioral sciences*, Ph.D. Thesis, Harvard University, Cambridge, MA, 1974.

[15] Saarinen S., R.B. Bramley and G. Cybenko, "Neural networks, backpropagation and automatic differentiation", *Automatic Differentiation of Algorithms: Theory, Implementation and Application*, eds. A. Griewank and G.F. Corliss, SIAM, Philadelphia, 1992, 31–42.

[16] Werbos P.J., *Backpropagation through time: What it does and how to do it.* V. 78, Proceedings of the IEEE, 1990, 1550–1560.

[17] Alekseev D.V., "Approximation of functions of several variables by neural networks", *Intelligent systems*, **7**:1-4 (2003), 191–205 (In Russian).

[18] Polovnikov V.S., *On optimization of the structural implementation of neural networks*, Ph.D. Thesis ... physical and mathematical sciences, MSU, Moscow, 2007 (In Russian).

[19] YAblonskij S.V., *Introduction to discrete mathematics*, eds. fiz.-mat.lit., «Science», Moscow, 1986 (In Russian).

[20] Haykin S., *Neural Networks. A Comprehensive Foundation*, Prentice Hall International, Inc., Canada, 1999.