

FUNCTION OF MULTIPLE VARIABLES AND NEURAL NETWORKS

A. N. GORBAN'

The following problems on the representation of function of multiple variables by means of simpler functions are discussed: are there functions of multiple variables which cannot be exactly represented by means of functions of one variable and when is an approximate representation of functions using simpler functions possible? An introduction to the theory of artificial neural networks is given. It is shown that with the aid of neural networks one can arbitrarily calculate precisely any continuous function.

Обсуждается представление функций многих переменных с помощью функций одного переменного. Дано введение в теорию искусственных нейронных сетей. Показано, что с помощью нейронных сетей можно сколь угодно точно вычислить любую непрерывную функцию.

© Горбань А.Н., 1998

ФУНКЦИИ МНОГИХ ПЕРЕМЕННЫХ И НЕЙРОННЫЕ СЕТИ

А. Н. ГОРБАНЬ

Красноярский государственный технический университет

СУЩЕСТВУЮТ ЛИ ФУНКЦИИ МНОГИХ ПЕРЕМЕННЫХ, КОТОРЫЕ НЕЛЬЗЯ ПРЕДСТАВИТЬ С ПОМОЩЬЮ ФУНКЦИЙ ОДНОГО ПЕРЕМЕННОГО?

Столь просто звучащий вопрос несет в себе много неясностей. Первая же попытка уточнения ставит в тупик. Где граница между существующим и несуществующим? Какие способы представления разрешены?

Попробуем пойти от очевидного. Зададимся вопросом: какие функции человек наверняка может вычислять? Если мы умеем складывать и умножать рациональные дроби (а это мы умеем), то можно точно вычислять многочлены и рациональные функции (отношения многочленов) с рациональными коэффициентами от рациональных же аргументов.

Можно, однако, задавать функции с помощью уравнений. Если считать решения нескольких простых уравнений известными, то класс вычисляемых функций расширится — решения некоторых более общих уравнений удастся выразить через эти более простые функции.

Классический пример: если использовать радикалы — решения уравнений $x^n = a$, то можно явно получить решения произвольных уравнений второй, третьей и четвертой степеней. Так, функция трех переменных a, b, c — решение уравнения $ax^2 + bx + c = 0$ — может быть точно выражена с помощью сложения, умножения, деления и функции одного переменного — квадратного корня. Эта формула знакома всем из курса алгебры средней школы. Намного более громоздкие и неизучаемые в школе формулы (но все же формулы с радикалами!) существуют для уравнений третьей и четвертой степени.

Вопрос: можно ли представить решение любого алгебраического уравнения с помощью радикалов — был окончательно и отрицательно решен Н.Х. Абелем и Э. Галуа в прошлом веке: уже уравнения пятой степени неразрешимы в радикалах.

Все же можно пытаться подбирать другие функции небольшого числа переменных — сложнее, чем радикалы, но проще, чем общие решения уравнений высоких степеней. Удастся ли с помощью этих функций построить решение любого уравнения? Вопрос был настолько важен, что Дж. Гильберт в списке своих проблем, которые, по его мнению, должны были определять развитие математики

XX века, под номером 13 поместил следующую задачу:

Представляется ли корень уравнения

$$x^7 + ax^3 + bx^2 + cx + 1 = 0$$

(как функция коэффициентов) суперпозицией каких-либо непрерывных функций двух переменных?

В приведенной формулировке неявно предлагается и ответ на вопрос: что означает “представить одну функцию с помощью других”? Он прост: это означает, что можно подставлять значения известных функций в другие известные функции в качестве аргументов (операция суперпозиции). В результате получаем новые функции, представленные с помощью известных. Так, например, функция пяти переменных

$$F(x_1, x_2, x_3, x_4, x_5) = f_1(f_1(x_1, x_2), f_2(x_3, f_1(x_4, x_5)))$$

представлена с помощью двух функций двух переменных f_1 и f_2 и операции суперпозиции (подстановки функции в функцию).

Для решений уравнений пятой и шестой степени такое представление возможно не только с помощью непрерывных, но даже аналитических функций. (Напомним, что аналитической в некоторой области D называется функция, которая в окрестности каждой точки D представима сходящимся степенным рядом. Аналитические функции имеют производные всех порядков и обладают еще многими “естественными” полезными свойствами. Среди представителей естественных наук встречается тенденция обращаться с каждой встречной функцией как с аналитической, использовать разложение в ряд и т.д. Увы, это не всегда возможно и может приводить к ошибкам.)

Оказалось полезным отвлечься от уравнений и поставить общий вопрос: можно ли произвольную непрерывную функцию n переменных получить с помощью операций сложения, умножения и суперпозиции из непрерывных функций двух переменных? Ответ оказался положительным! В серии работ А.Н. Колмогоров, затем В.И. Арнольд и вновь А.Н. Колмогоров [1, 3] решили эту проблему: можно получить любую непрерывную функцию n переменных с помощью операций сложения, умножения и суперпозиции из непрерывных функций одного переменного.

Последняя теорема А.Н. Колмогорова из этой серии настолько проста и изящна, что мы чуть позже приведем ее формулировку целиком. А пока несколько замечаний об условиях теоремы.

От условия непрерывности можно отказаться, тогда получится довольно тривиальный результат, связанный по существу с равносильностью отрезка и куба любой размерности. Условие непрерывности нельзя значительно усилить: уже существование производных (дифференцируемость) существенно осложняет дело. Нельзя представить все l раз непре-

рывно дифференцируемые функции трех переменных в виде суперпозиций функций двух переменных, каждая из которых дифференцируема $[2l/3] + 1$ раз (выражение $[2l/3]$ означает целую часть числа $2l/3$). Это доказано А.Г. Витушкиным [2].

А теперь теорема, завершившая серию исследований для непрерывных функций.

Теорема Колмогорова. *Каждая непрерывная функция n переменных, заданная на единичном кубе n -мерного пространства, представима в виде*

$$f(x_1, x_2, \dots, x_n) = \sum_{q=1}^{2n+1} h_q \left[\sum_{p=1}^n \varphi_q^p(x_p) \right], \quad (1)$$

где функции $h_q(u)$ непрерывны, а функции $\varphi_q^p(x_p)$, кроме того, еще и стандартны, то есть не зависят от выбора функции f .

В частности, каждая непрерывная функция двух переменных x, y представима в виде

$$f(x, y) = \sum_{q=1}^5 h_q[\varphi_q(x) + \psi_q(y)]. \quad (2)$$

Эта теорема не только элегантна, но и проста. Ее доказательство элементарно изложено в блестящей статье В.И. Арнольда [3].

Обсудим содержание теоремы. В n -мерном пространстве вводится $2n + 1$ специальных (и весьма экзотических) координат (на плоскости их пять). Они являются функциями вида

$$\sum_{p=1}^n \varphi_q^p(x_p) \quad (3)$$

при различных $q = 1, 2, \dots, 2n + 1$ от обычных координат x_1, x_2, \dots, x_n . Любая непрерывная функция представляется как сумма непрерывных функций от отдельных координат, и в эту сумму входит не более одной функции для каждой координаты (3).

Всюду речь идет именно о непрерывных функциях – таких гладких координатных функций не существует, функции $\varphi_q^p(x_p)$, построенные Колмогоровым, не имеют производных, хотя и непрерывны (с точки зрения обычной физической интуиции это функции-монстры, которые трудно себе представить).

Так существуют ли функции многих переменных? В каком-то смысле нет: в классе непрерывных функций достаточно использовать только функции одного переменного, суперпозицию функций и сложение и за конечное число операций мы получим точное выражение любой функции многих переменных. Для более узкого класса дифференцируемых функций это уже не так.

ПРИБЛИЖЕННОЕ ПРЕДСТАВЛЕНИЕ ФУНКЦИЙ С ПОМОЩЬЮ БОЛЕЕ ПРОСТЫХ: ТЕОРЕМЫ ВЕЙЕРШТРАССА И СТОУНА

В 13-й проблеме Гильберта и теореме Колмогорова речь шла о точном представлении функций многих переменных с помощью функций одного переменного. Оказалось, что в классе непрерывных функций такое представление возможно. Но кроме вопроса о точном представлении существует еще один – об аппроксимации. Можно даже предположить, что он важнее – вычисление большинства функций производится приближенно даже при наличии точных формул.

Приближение функций многочленами и рациональными функциями имеет историю, еще более давнюю, чем проблема точного представления. Знаменитая теорема Вейерштрасса утверждает, что непрерывную функцию скольких переменных $f(x_1, x_2, \dots, x_n)$ на замкнутом ограниченном множестве Q можно равномерно приблизить последовательностью полиномов: для любого $\varepsilon > 0$ существует такой многочлен $P(x_1, x_2, \dots, x_n)$, что максимум его отклонения от $f(x_1, x_2, \dots, x_n)$ на Q не превосходит данного ε :

$$\max_Q |f(x_1, x_2, \dots, x_n) - P(x_1, x_2, \dots, x_n)| < \varepsilon.$$

Примечание. Здесь и далее мы будем использовать выражения “замкнутое ограниченное множество” и “компактное пространство X ”. Читатель, который не очень знаком с этими понятиями, может представлять себе просто n -мерный прямоугольный параллелепипед – множество наборов переменных (x_1, x_2, \dots, x_n) , удовлетворяющих неравенствам $a_i \leq x_i \leq b_i$ для некоторых чисел $a_i < b_i$.

Чтобы сформулировать обобщения и усиления теоремы Вейерштрасса, необходимо перейти к несколько более абстрактному языку. Рассмотрим компактное пространство X и множество $C(X)$ непрерывных функций на X с вещественными значениями. Произведение непрерывной функции на число, суммы и произведения непрерывных функций также являются таковыми. Это означает, что $C(X)$ образует алгебру.

Если X – n -мерный прямоугольный параллелепипед, то теорема Вейерштрасса о равномерном приближении функций многочленами может быть сформулирована так. Пусть $E \subseteq C(X)$ – подалгебра в $C(X)$, $1 \in E$ и координатные функции ($f_i \equiv x_i$) принадлежат E . Тогда E плотно в $C(X)$, то есть любая непрерывная функция на X может быть сколь угодно точно равномерно приближена элементами из E .

Действительно, E – подалгебра, то есть суммы и произведения элементов E принадлежат E . Координатные функции принадлежат E , следовательно, любые их суммы, произведения, суммы произведений, то есть все многочлены, принадлежат E . По теореме Вейерштрасса это E плотно в $C(X)$.

Пусть далее X – произвольное компактное пространство. Сильным обобщением теоремы о возможности равномерного приближения непрерывных функций многочленами является

Теорема Стоуна [4, 5]. Пусть $E \subseteq C(X)$ – подалгебра в $C(X)$, $1 \in E$ и функции из E разделяют точки в X (то есть для любых различных $x, y \in X$ существует такая функция $g \in E$, что $g(x) \neq g(y)$). Тогда E плотно в $C(X)$, то есть любая непрерывная функция на X может быть сколь угодно точно равномерно приближена элементами из E .

Теорема Стоуна обобщает теорему Вейерштрасса по двум направлениям. Во-первых, рассматриваются функции на произвольном компакте, но не это самое важное. Во-вторых, доказано утверждение, новое даже для функций одного переменного (не говоря уже о многих): плотно не только множество многочленов от координатных функций, но вообще кольцо многочленов от любого набора функций, разделяющих точки. Следовательно, например, плотно множество тригонометрических многочленов.

А вот и другой пример, очень важный для многих приложений. Рассмотрим всюду отрицательные (неположительные) многочлены второго порядка от нескольких переменных $Q(x)$. Сумма таких многочленов тоже отрицательный (неположительный) многочлен второго порядка. Пусть E – множество линейных комбинаций вида

$$A_1 e^{Q_1(x)} + A_2 e^{Q_2(x)} + \dots + A_k e^{Q_k(x)}, \quad (4)$$

где A_i – числа, e – основание натуральных логарифмов.

Легко проверить, что E – подалгебра в $C(X)$, а множество функций из E вида

$$e^{-(x_1 - a_1)^2 - \dots - (x_n - a_n)^2}$$

при всевозможных a_i разделяет точки в n -мерном пространстве (значение такой функции в точке a больше, чем во всех других точках). Поэтому множество линейных комбинаций (4) плотно среди непрерывных функций на любом ограниченном замкнутом множестве.

Теорема Стоуна дает рецепт конструирования конкретных обобщений теоремы Вейерштрасса: достаточно взять произвольный набор функций, разделяющих точки, построить все многочлены от них и получится плотное в $C(X)$ множество функций.

ПРЕДСТАВЛЕНИЕ ФУНКЦИЙ С ПОМОЩЬЮ НЕЙРОННЫХ СЕТЕЙ

Кроме аппроксимации функций многочленами и их обобщениями из колец функций, разделяющих точки, в последнее время все большее внимание уделяется приближению функций многих переменных с помощью линейных операций и суперпозиций

функций одного переменного. Такое приближение осуществляется специальными формальными устройствами – нейронными сетями. Каждая сеть состоит из формальных нейронов. Нейрон получает на входе вектор сигналов x , вычисляет его скалярное произведение на вектор весов α и некоторую функцию одного переменного $\varphi(x, \alpha)$. Результат рассылается на входы других нейронов или передается на выход. Таким образом, нейронные сети вычисляют суперпозиции простых функций одного переменного и их линейных комбинаций.

Одна из наиболее распространенных задач, решаемых с помощью нейронных сетей, состоит в оценивании и аппроксимации по нескольким точкам (примерам) функций вход–выход для решения в дальнейшем задачи прогноза неизвестных значений выходов по известным входам.

Актуальным становится вопрос о том, какие функции могут быть аппроксимированы с их помощью.

Для описания алгоритмов и устройств в нейронной информатике выработана специальная схемотехника, в которой элементарные устройства – сумматоры, синапсы, нейроны и т.п. – объединяются в сети, предназначенные для решения задач.

Интересен статус этой схемотехники. Для многих начинающих кажется неожиданным, что ни в аппаратной реализации нейронных сетей, ни в профессиональном программном обеспечении все эти элементы вовсе не обязательно реализуются как отдельные части или блоки. Используемая в нейронной информатике идеальная схемотехника представляет собой особый язык для представления нейронных сетей и их обсуждения. При программной и аппаратной реализации выполненные на этом языке описания переводятся на языки другого уровня, более пригодные для реализации.

Самый заслуженный и, вероятно, наиболее важный элемент нейросистем – это **адаптивный сумматор**. Он вычисляет скалярное произведение вектора входного сигнала x на вектор параметров α . На схемах будем обозначать его так, как показано на рис. 1. Адаптивным называем его из-за наличия вектора настраиваемых параметров α . Для многих задач полезно иметь линейную неоднородную функцию выходных сигналов. Ее вычисление также можно представить с помощью адаптивного сумматора, имеющего $n + 1$ вход и получающего на 0-й вход постоянный единичный сигнал (рис. 2).

Нелинейный преобразователь сигнала изображен на рис. 3. Он получает скалярный входной сигнал x и переводит его в $\varphi(x)$.

Точка ветвления служит для рассылки одного сигнала по нескольким адресам. Она получает скалярный входной сигнал x и передает его всем своим выходам. **Стандартный формальный нейрон** составлен из входного сумматора, нелинейного преобразователя и точки ветвления на выходе (рис. 4).

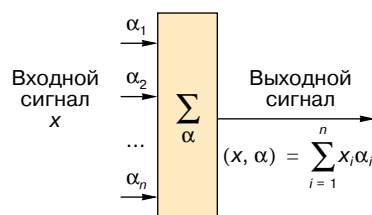


Рис. 1. Адаптивный сумматор

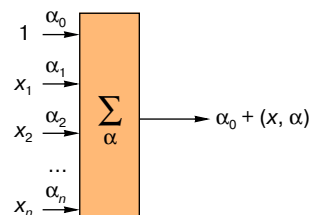


Рис. 2. Неоднородный адаптивный сумматор



Рис. 3. Нелинейный преобразователь сигнала

Линейная связь – синапс отдельно от сумматоров не встречается, однако для некоторых рассуждений бывает удобно выделить этот элемент. Он умножает входной сигнал x на “вес синапса” α .

Итак, дано описание основных элементов, из которых составляются нейронные сети. Перейдем теперь к вопросу: как можно составлять эти сети? Строго говоря, как угодно, лишь бы входы получали какие-нибудь сигналы. Но такой произвол необозрим, поэтому используют несколько стандартных архитектур, из которых путем вырезания лишнего или (реже) добавления строят большинство используемых сетей.

Сначала следует договориться о том, как будет согласована работа различных нейронов во времени. Как только в системе возникает более одного

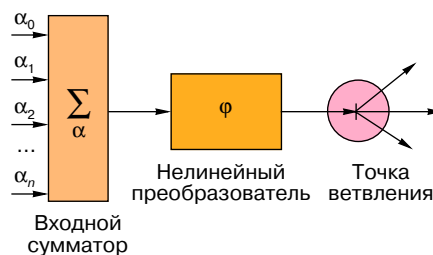


Рис. 4. Формальный нейрон

элемента, встает вопрос о синхронности функционирования. Для привычных нам всем программных имитаторов нейронных сетей на цифровых ЭВМ такого вопроса нет только из-за свойств основного компьютера, на котором реализуются нейронные сети. Для других способов реализации такой вопрос весьма важен. Все же здесь и далее рассматриваются только нейронные сети, синхронно функционирующие в дискретные моменты времени: все нейроны срабатывают разом.

В зоопарке нейронных сетей можно выделить две базовые архитектуры: слоистые и полносвязные сети.

Слоистые сети. Нейроны расположены в несколько слоев (рис. 5). Нейроны первого слоя получают входные сигналы, преобразуют их и через точки ветвления передают нейронам второго слоя. Далее срабатывает второй слой и т.д. до k -го слоя, который выдает выходные сигналы для интерпретатора и пользователя. Если не оговорено противное, то каждый выходной сигнал i -го слоя подается на вход всех нейронов $(i + 1)$ -го. Число нейронов в каждом слое может быть любым и никак заранее не связано с количеством нейронов в других слоях. Стандартный способ подачи входных сигналов: все нейроны первого слоя получают каждый входной сигнал. Особое распространение получили трехслойные сети, в которых каждый слой имеет свое наименование: первый – входной, второй – скрытый, третий – выходной.

Полносвязные сети. Каждый нейрон передает свой выходной сигнал остальным нейронам, включая самого себя. Выходными сигналами сети могут быть все или некоторые выходные сигналы нейронов после нескольких тактов функционирования сети. Все входные сигналы подаются всем нейронам.

Элементы слоистых и полносвязных сетей могут выбираться по-разному. Существует, впрочем, стандартный выбор – нейрон с адаптивным неоднородным линейным сумматором на входе (см. рис. 4).

Для полносвязной сети входной сумматор нейрона фактически распадается на два: первый вычисляет линейную функцию от входных сигналов сети, второй – линейную функцию от выходных сигналов других нейронов, полученных на предыдущем шаге.

Нейронные сети вычисляют линейные функции, нелинейные функции одного переменного, а также всевозможные суперпозиции – функции от функций, получаемые при каскадном соединении сетей. Что можно получить, используя только такие операции? Какие функции удастся вычислить точно, а какие функции можно сколь угодно точно аппроксимировать с помощью нейронных сетей? Чтобы изучить возможности нейронных сетей, нужно ответить на эти вопросы [6, 7].

Каждый выходной сигнал нейронной сети будем рассматривать как вычисляемую этой сетью функцию от ее входных сигналов. Пусть задана функция активации нейронов (характеристическая функция) ϕ – нелинейный преобразователь, преобразующий выходной сигнал сумматора (см. рис. 4). Задан вопрос: какие функции могут быть вычислены сетями с данной функцией активации. Множество этих функций обладает следующими свойствами: оно является линейным пространством (так как можно вычислить линейную комбинацию выходных сигналов нескольких сетей с помощью дополнительного сумматора), содержит константы и координатные функции, а также вместе с любой функцией f и суперпозицию $\phi(f)$. Оказывается, что отсюда следует, что множество таких функций плотно среди всех непрерывных функций на любом замкнутом ограниченном множестве. Это доказывается на основе следующей обобщенной теоремы Стоуна.

Пусть $E \subseteq C(X)$ – линейное пространство, $C(\mathbf{R})$ – пространство непрерывных функций на действительной оси \mathbf{R} , $f \in C(\mathbf{R})$ – нелинейная функция и для любого $g \in E$ выполнено $f(g) \in E$. В этом случае

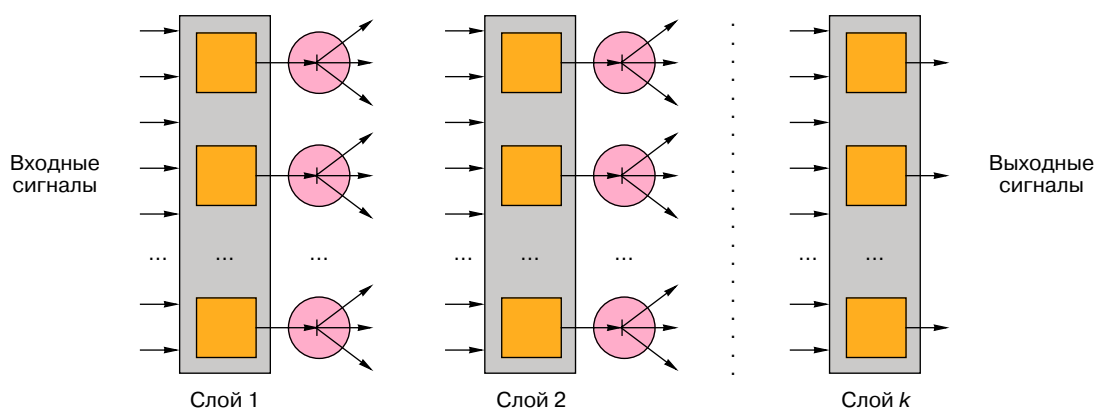


Рис. 5. Слоистая сеть

будем говорить, что E замкнуто относительно нелинейной унарной операции f .

Очевидный пример: множество функций n переменных, которые можно точно представить, используя заданную функцию f одного переменного и линейные функции, является линейным пространством, замкнутым относительно нелинейной унарной операции f .

Замечание. Линейное пространство $E \subseteq C(X)$ замкнуто относительно нелинейной операции $f(x) = x^2$ тогда и только тогда, когда E является подалгеброй в $C(X)$.

Действительно, $fg = \frac{1}{2}[(f+g)^2 - f^2 - g^2]$, поэтому для линейного пространства $E \subseteq C(X)$ замкнутость относительно унарной операции $f(x) = x^2$ равносильна замкнутости относительно произведения функций.

Согласно приведенному замечанию, теорема Стоуна может быть переформулирована так.

Пусть $E \subseteq C(X)$ – линейное подпространство в $C(X)$, $1 \in E$, функции из E разделяют точки в X и E замкнуто относительно нелинейной унарной операции $f(x) = x^2$. Тогда E плотно в $C(X)$.

Требуемое обобщение теоремы Стоуна состоит в замене $f(x) = x^2$ на произвольную нелинейную непрерывную функцию.

Обобщенная аппроксимационная теорема. Пусть $E \subseteq C(X)$ – линейное подпространство в $C(X)$, $1 \in E$, функции из E разделяют точки в X и E замкнуто относительно нелинейной унарной операции $f \in C(\mathbf{R})$. Тогда E плотно в $C(X)$.

Эту теорему можно трактовать как утверждение об универсальных аппроксимационных свойствах любой нелинейности: с помощью линейных операций и каскадного соединения можно из произвольных нелинейных элементов получить любой требуемый результат с любой наперед заданной точностью [7].

В доказательстве есть только одна нетривиальная идея – рассмотреть все нелинейные операции $f \in C(\mathbf{R})$, относительно которых замкнуто E , и показать, что это множество операций совпадает со всем пространством непрерывных функций. Дальнейшее – дело техники. Любопытно, что исследование таких множеств операций было начато более 35 лет назад [8]¹ вне какой-либо связи с задачами приближения функций – поучительный пример потенциальной пользы абстрактных математических конструкций.

Главный вопрос этого раздела: что могут вычислять нейронные сети. Ответ получен: в каком-то смысле нейронные сети могут все. Остается открытым другой вопрос: как их этому научить?

¹ Автор признателен за эту ссылку редакции журнала “Функциональный анализ и его приложений”.

ДВА СЛОВА ОБ ОБУЧЕНИИ

Обучению нейронных сетей уже посвящены тысячи работ. Известны введения в теорию этого обучения объемом по 600–700 страниц. Но для первого знакомства достаточно, быть может, и пары страниц.

Примерно половина обучаемых нейронных сетей использует “задачник с решениями” – набор примеров с известными правильными ответами. Каждый такой пример представляет собой пару вход – известный выход. Входные сигналы из этих пар подаются обучаемой сети на обработку, а получаемые выходные сигналы сравниваются с ответом – строится оценка работы сети; для определенности принимаем: чем меньше оценка, тем лучше работает сеть. Далее функция оценки шаг за шагом минимизируется как функция параметров – синаптических весов. Процесс ее пошаговой минимизации устроен так:

1) начинаем с некоторой (чаще всего случайной) начальной точки;

2) ищем такое направление в пространстве параметров, в котором функция оценки убывает (направление спуска – такое название возникло из уподобления минимизируемой функции высоте, а минимизации – спуску в яму);

3) шагаем в этом направлении, пока функция убывает, затем либо возвращаемся к поиску нового направления, либо заканчиваем обучение.

Успех метода во многом связан с определением направления спуска. Простейший и самый популярный выбор – вектор g антиградиента функции оценки. Он называется направлением наискорейшего спуска. Если обозначить функцию оценки H , а набор параметров $\alpha_1, \alpha_2, \dots, \alpha_n$, то координаты антиградиента $g_i = -\partial H / \partial \alpha_i$ (со знаком минус берутся производные функции оценки по соответствующим настраиваемым параметрам). Существует множество способов улучшить это направление, но наискорейший спуск хорош хотя бы тем, что он действительно спуск – вдоль антиградиента функция убывает.

Получим формулы для обучения адаптивного сумматора методом наискорейшего спуска. Сумматор вычисляет скалярное произведение вектора сигналов x на вектор весов α_i :

$$y = (\alpha, x) = \sum_i \alpha_i x_i.$$

Пусть для вектора входных сигналов x требуется получить выходной сигнал y^* . Определим функцию оценки: $H = \frac{1}{2}(y^* - y)^2$ – половина квадрата отклонения выходного сигнала от его требуемого значения. Направление наискорейшего спуска: $g_i = (y^* - y)x_i$ – произведение ошибки $y^* - y$ на сигнал x_i . Мы получили знаменитую формулу Уидроу, на

основании которой построены многие обучающиеся устройства [9].

Если задачник содержит более одного примера, то можно складывать соответствующие оценки и направления спуска.

Получив направление, надо по нему шагать — заменять вектор α на $\alpha + hg$. Какой шаг h выбрать? Простейший рецепт: если шаг удачен (оценка уменьшилась), продолжаем движение, каждый раз удваивая шаг, пока улучшение не прекратится. Останавливаемся на последнем удачном шаге. Если шаг неудачен (оценка увеличилась), делаем из той же начальной точки половинный пробный шаг, далее снова его дробим и т.д., пока оценка не станет лучше исходной. Далее продолжаем дробить шаг, пока оценка не начнет увеличиваться. Останавливаемся в той точке, где оценка была минимальной.

Для первого представления об обучении сказанного достаточно. Надо еще только заметить, что вычисление всего антиградиента оценки (n координат вектора g) всего лишь в 2–3 раза сложнее, чем вычисление одной из его координат. Это очень простое, но неожиданное открытие и вызвало бурное развитие методов пошагового обучения нейронных сетей, но о нем в другой статье.

НЕЙРОБУМ: МОДА В НАУКЕ И НЕЙРОННЫЕ СЕТИ

В словах “искусственные нейронные сети” слышатся отзвуки фантазий об андроидах и бунте роботов, о машинах, заменяющих и имитирующих человека. Эти фантазии интенсивно поддерживаются многими разработчиками нейросистем: рисуется не очень отдаленное будущее, в котором роботы осваивают различные виды работ, просто наблюдая за человеком, а в более отдаленной перспективе человеческое сознание и личность перегружаются в искусственную нейронную сеть — появляются шансы на вечную жизнь.

Нейронные сети нынче в моде, и поэтическая реклама делает свое дело, привлекает внимание. Но стоит ли следовать за модой? Ресурсы ограничены — особенно у нас, особенно теперь. Все равно всего на всех не хватит. И возникают вопросы:

1) нейрокомпьютер — это интеллектуальная игрушка или новая техническая революция?

2) что нового и полезного может сделать нейрокомпьютер?

За этими вопросами скрыты два базовых предположения:

1) на новые игрушки, даже высокоинтеллектуальные, средств нет;

2) нейрокомпьютер должен доказать свои новые возможности — сделать то, чего не может сделать обычная ЭВМ, иначе на него не стоит тратить.

У энтузиастов есть свои рекламные способы отвечать на заданные вопросы, рисуя светлые после-

завтрашние горизонты. Но все это в будущем. А сейчас? Ответы парадоксальны:

1) нейрокомпьютеры — это новая техническая революция, которая приходит к нам в виде интеллектуальной игрушки (вспомните: и персональные ЭВМ были придуманы для игры!);

2) для любой задачи, которую может решить нейрокомпьютер, можно построить более стандартную специализированную ЭВМ, которая решит ее не хуже, а чаще всего даже лучше.

Зачем же тогда нейрокомпьютеры? Вступая в творческую игру, мы не можем знать, чем она кончится, иначе это не Игра. Поэзия и реклама дают нам фантом, призрак результата, погоня за которым — важнейшая часть игры. Столь же призрачными могут оказаться и прозаичные ответы — игра может далеко от них увести. Но и они необходимы — трудно бегать по облакам и иллюзия практичности столь же важна, сколь и иллюзия величия. Вот несколько вариантов прозаичных ответов на вопрос “зачем?” — можно выбрать, что для вас важнее.

А. Нейрокомпьютеры дают стандартный способ решения многих нестандартных задач. И неважно, что специализированная машина лучше решит один класс задач. Важнее, что один нейрокомпьютер решит и эту задачу, и другую, и третью, и не надо каждый раз проектировать специализированную ЭВМ — нейрокомпьютер сделает все сам и почти не хуже.

Б. Вместо программирования — обучение. Нейрокомпьютер учится — нужно только формировать учебные задачки. Труд программиста замещается новым трудом — учителя (может быть, надо сказать, тренера или дрессировщика). Лучше это или хуже? Ни то ни другое. Программист предписывает машине все детали работы, учитель создает “образовательную среду”, к которой приспосабливается нейрокомпьютер. Появляются новые возможности для работы¹.

В. Нейрокомпьютеры особенно эффективны там, где нужно подобие человеческой интуиции — для распознавания образов (узнавания лиц, чтения рукописных текстов), перевода с одного естественного языка на другой и т.п. Именно для таких задач обычно трудно сочинить явный алгоритм.

Г. Гибкость структуры: можно различными способами комбинировать простые составляющие нейрокомпьютеров — нейроны и связи между ними. За счет этого на одной элементной базе и даже внутри “тела” одного нейрокомпьютера можно создавать совершенно различные машины. Появляется еще одна новая профессия — нейроконструктор (конструктор мозгов).

¹ “Обучение машин” задачам распознавания много старше, чем нейрокомпьютеры. Как это часто бывает, важнейшие идеи переходят из области в область, а момент их первого рождения скрыт во времени.

Д. Нейронные сети позволяют создать эффективное программное обеспечение для высокопараллельных компьютеров. Для высокопараллельных машин хорошо известна проблема, как их эффективно использовать, как добиться, чтобы все элементы одновременно и без лишнего дублирования вычисляли что-нибудь полезное? Создавая математическое обеспечение на базе нейронных сетей можно для широкого класса задач решить эту проблему.

Если перейти к еще более прозаическому уровню повседневной работы, то нейронные сети — это всего-навсего сети, состоящие из связанных между собой простых элементов — формальных нейронов. Значительная часть работ по нейроинформатике посвящена переносу различных алгоритмов решения задач на такие сети.

Ядром используемых представлений является идея о том, что нейроны можно моделировать довольно простыми автоматами, а вся сложность мозга, гибкость его функционирования и другие важнейшие качества определяются связями между нейронами. Каждая связь представляется как совсем простой элемент, служащий для передачи сигнала. Предельным выражением этой точки зрения может служить лозунг “структура связей — все, свойства элементов — ничто”.

Обобщенная аппроксимационная теорема об универсальных свойствах любой нелинейности является одним из математических оснований для коннекционизма: *неважно, какая нелинейная функция используется в нейроне, существенны лишь факт нелинейности и свобода в построении системы связей.*

ЛИТЕРАТУРА

1. Колмогоров А.Н. О представлении непрерывных функций нескольких переменных в виде суперпозиции непрерывных функций одного переменного // Докл. АН СССР. 1957. Т. 114, № 5. С. 953–956.
2. Витушкин А.Г. О многомерных вариациях. М.: Физматгиз, 1955.
3. Арнольд В.И. О представлении функций нескольких переменных в виде суперпозиции функций меньшего числа переменных // Мат. просвещение. 1958. Вып. 3. С. 41–61.
4. Stone M.N. The Generalized Weierstrass Approximation Theorem // Math. Mag. 1948. Vol. 21. P. 167–183, 237–254.
5. Шефер Х. Топологические векторные пространства. М.: Мир, 1971.
6. Subenko G. Approximation by Superposition of a Sigmoidal Function // Mathematics of Control, Signals and Systems. 1989. Vol. 2. P. 303–314.
7. Горбань А.Н., Россиев Д.А. Нейронные сети на персональном компьютере. Новосибирск: Наука, 1996.
8. Leeuw K., Katznelson Y. Functions that Operate on Non-Self-Adjoint Algebras // J. d'Anal. Math. 1963. Vol. 11. P. 207–219.
9. Уидроу Б., Стирнз С. Адаптивная обработка сигналов. М.: Мир, 1989. 440 с.

* * *

Александр Николаевич Горбань, доктор физико-математических наук, профессор, зав. кафедрой нейро-ЭВМ Красноярского государственного технического университета и лабораторией моделирования неравновесных систем Красноярского вычислительного центра Сибирского отделения РАН. Области научных интересов — динамика, математическая физика, нейронные сети. Автор более 100 научных работ и 11 монографий.