

М.П. Лазарев<sup>1</sup>, И.В. Яковлев<sup>1</sup>, Байкова Л.А.<sup>1</sup>

<sup>1</sup> Иркутский государственный университет путей сообщения, г. Иркутск, Российская Федерация

## ПРИМЕНЕНИЕ ЭЛЕМЕНТОВ ТЕОРИИ ПОЛЯ В ГЛУБОКОМ МАШИННОМ ОБУЧЕНИИ

**Аннотация.** На сегодняшний день понятие машинного обучения играет немаловажную роль в развитии современных технологий, в том числе, систематизации и обработке информации. Целью данного исследования является разъяснение основных понятий, а также демонстрация применения математических методов вычисления в этой отрасли. В результате исследования будут даны определения со ссылками на источники ключевых понятий и продемонстрировано наглядное решение типовой задачи с применением математических методов оптимизации выдаваемых результатов.

**Ключевые слова:** машинное обучение, искусственные нейронные сети, глубокое машинное обучение, искусственный интеллект.

M. P. Lazarev<sup>1</sup>, I. V. Yakovlev<sup>1</sup>, L. A. Baykova<sup>1</sup>

<sup>1</sup> Irkutsk State Transport University, Irkutsk, Russia

## APPLICATION OF FIELD THEORY ELEMENTS IN DEEP MACHINE LEARNING

**Abstract.** Today, the concept of machine learning plays an important role in the development of modern technologies, including systematization and information processing. The purpose of this study is to clarify the basic concepts, as well as to demonstrate the application of mathematical methods of calculation in this industry. As a result of the research, definitions will be given with references to the sources of keywords. A visual solution to a typical problem will be demonstrated using mathematical methods to optimize the results.

**Keywords:** machine learning, artificial neural networks, deep machine learning, artificial intelligence.

### Введение

На сегодняшний день такие понятия как глубокое машинное обучение, искусственный интеллект и искусственные нейронные сети не просто являются отголосками потенциального будущего, а уже сейчас очень важны и актуальны для многих отраслей информационных технологий в сферах обслуживания, упрощения взаимодействия с пользователем. Областей, где задействованы технологии машинного обучения огромное количество. Так, например, сегодня распространены технологии подтверждения личности посредством радужки глаза («По данным World Payments Report от 2017 года в период с 2015-2019 гг. ожидаемый рост доли платежных транзакций, осуществляемых с их помощью, составит 21.8%, и 32% в период 2017-2022 гг.[1, с. 2]) или же настройка таргетированной выдачи в новостных лентах различных социальных сетей на основе предпочтений пользователя и т.д.

Возможно возникнет сомнение насчет необходимости в развитии столь сложных технологий, т.к. подобные задачи человек сможет выполнять с идентичной результативностью (а в редких ситуациях даже лучше), однако, из-за увеличения числа пользователей продуктов информационных технологий, а, следовательно, и увеличения объемов обрабатываемой информации, сложно обойтись без автоматизации. Впрочем, резонно также задаться вопросом: неужели невозможно обойтись общеизвестными регламентированными правилами на основе логических операторов в программировании - «if» и «else»? Так, например, когда только появлялись системы «интеллектуальных» приложений по типу электронной почты, для нахождения и классификации сообщений определенного содержания применялись простые правила: если (if) в письме присутствует запрещенный список слов (например, «Распродажа!», «Скидки» и пр.), то письмо помещалось в папку «Спам»; иначе (else) – во входящие. Однако, данный подход имеет два существенных недостатка, первым из которых является логика

принятия решений, относящаяся исключительно к одной конкретной области (т.е. незначительное изменение задачи влечет за собой потенциальную модернизацию всей системы); второй недостаток заключается в том, что при проектировании подобных, казалось бы, простых правил требуется углубленное понимание всей системы. Очевидная ситуация, при которой данный подход будет бесполезен - это распознавание лиц на изображениях, т.к. невозможно задать настолько сложные логические условия. Данная возможность на сегодняшний день имеется в каждом смартфоне именно благодаря машинному обучению [2, с. 13].

### История вопроса

Вероятно, уместность этого раздела в статье покажется сомнительной ввиду кажущейся новизны данной темы, но в действительности глубокое машинное обучение возникло в 1940 – х годах. Модернизм этого понятия обусловлен тем, что в течение многих лет оно было забыто и часто меняло названия и собственную цель разработки.

Было три волны разработок в данном направлении: в 1940-1960-е года глубокое обучение носило название «Кибернетика», в 1980-1990 – коннекционизм, современное название присвоено с 2006-го года.

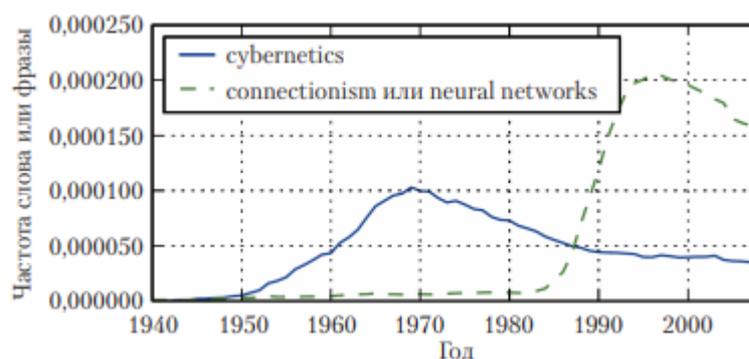


Рис. 1. На графике показаны две из трех исторических волн исследований по искусственным нейронным сетям. График составлен по частоте фраз «cybernetics» и «connectionism или neural networks» согласно Google Books (третья волна началась недавно и еще не отражена)

Так, в первой волне разрабатывались теории биологического обучения: получилось обучить один нейрон. Вторая волна обусловлена возникновением метода обратного распространения с применением к обучению нейронной сети с несколькими скрытыми слоями. Третья нынешняя волна отличается от предыдущих тем, что использует в обучении более общий подход, который не обязательно должен коррелировать с устройством биологического мозга, а также ставит собственной целью решение задач обработки больших баз данных, чьим возникновением и обязана [3, с. 31].

### Основные понятия.

Нейронная сеть – это структура нейронов, соединенных между собой синапсами. Данная система пришла в программирование из биологии и позволяет компьютеру обрести способность запоминать, анализировать и воспроизводить информацию из памяти. Основная цель искусственных нейронных сетей заключается в решении трудных задач, требующих аналитических преобразований на подобие тех, что способен произвести человеческий мозг.

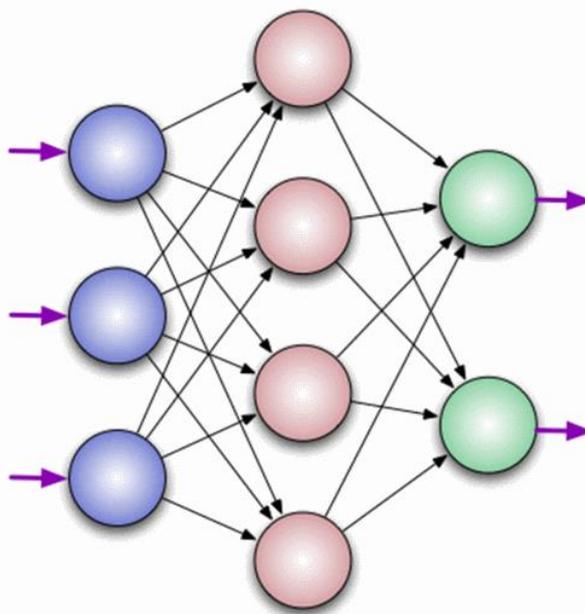
Так, самыми распространенными задачами искусственных нейронных сетей являются:

- Классификация – распределение данных по параметрам.
- Предсказание – возможность предсказывать следующий шаг.
- Распознавание – на данный момент решение подобного рода задач пользуется максимальным спросом. Например, нахождение вашего лица при использовании камеры смартфона.

Определение нейрона дается в учебном пособии: «Нейрон представляет из себя элемент, который вычисляет выходной сигнал (по определенному правилу) из совокупности входных сигналов. То есть

основная последовательность действий одного нейрона такая:

- Прием сигналов от предыдущих элементов сети
- Комбинирование входных сигналов
- Вычисление выходного сигнала
- Передача выходного сигнала следующим элементам нейронной сети [4. с. 6].



**Рис.2. Упрощенная схема нейронной сети: синие – входные, красные – скрытые, зеленые – выходные**

Входные нейроны, скрытые и выходные – все они составляют структуры нейронной сети, если не включать контекстные и скрытые нейроны, о которых речь пойдет далее в статье.

Термин слоя вводят, если используется большое количество нейронов. Название слоев соответствует функции нейронов (например, входные нейроны соответствуют входному слою).

Все нейроны обладают входным и выходным параметром (input/output data). У входного нейрона  $input=output$ . В input попадают суммированные входные данные нейронов предыдущих слоев. Впоследствии данная информация в нейроне преобразуется посредством функции активации и, соответственно, попадает в поле output.

Синапс – это связь между двумя нейронами, имеющая только вес в качестве параметра. Именно благодаря этому параметру входная информация изменяется при передаче между нейронами. Также, если в один нейрон соединен с несколькими нейронами из предыдущего слоя, то передаваемая информация будет приоритетней у нейрона с соответствующим ему большим весом. Во время инициализации нейронной сети веса расставляются в случайном порядке.

Следует отметить, что входные значения необходимо нормализовать ввиду того, что нейроны способны оперировать числами в диапазоне  $[0;1]$  или  $[-1;1]$ . Нормализовать значение следует посредством функции активации.

Функция активации, таким образом, это способ нормализации входных данных. Существует множество различных функций активаций. В качестве примера будут приведены самые популярные: линейная функция (рис. 3), сигмоид (рис. 4) и гиперболический тангенс (рис. 5).

$$f(x) = x$$

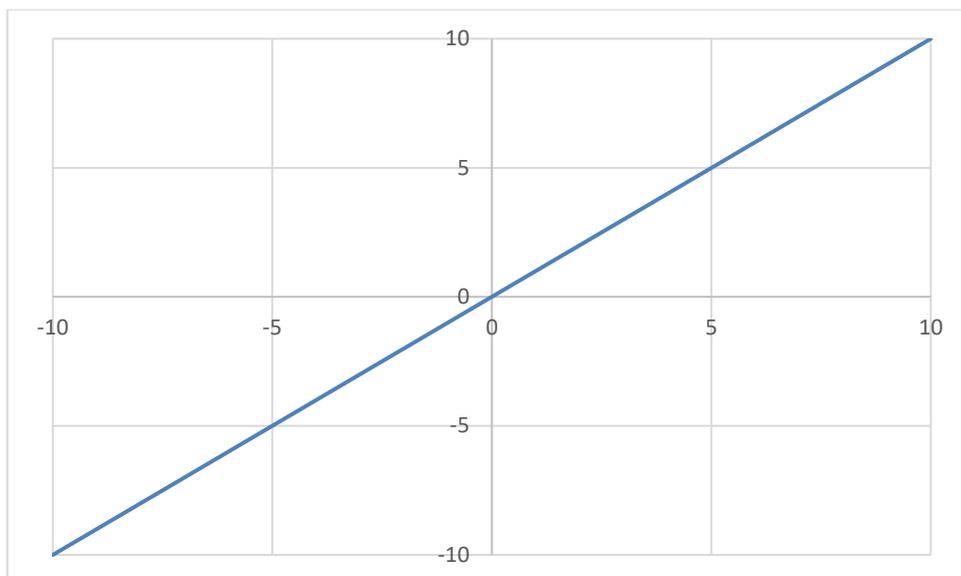


Рис. 3. Линейная функция

$$f(x) = \frac{1}{1 + e^{-x}}$$

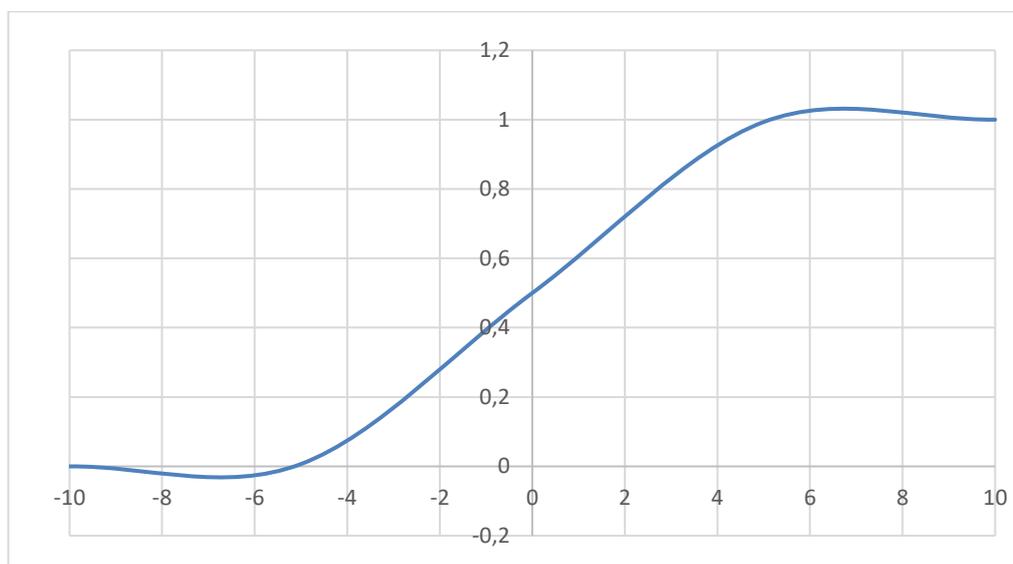


Рис. 4. Сигмоида

$$f(x) = \frac{e^{2x} - 1}{e^{2x} + 1}$$

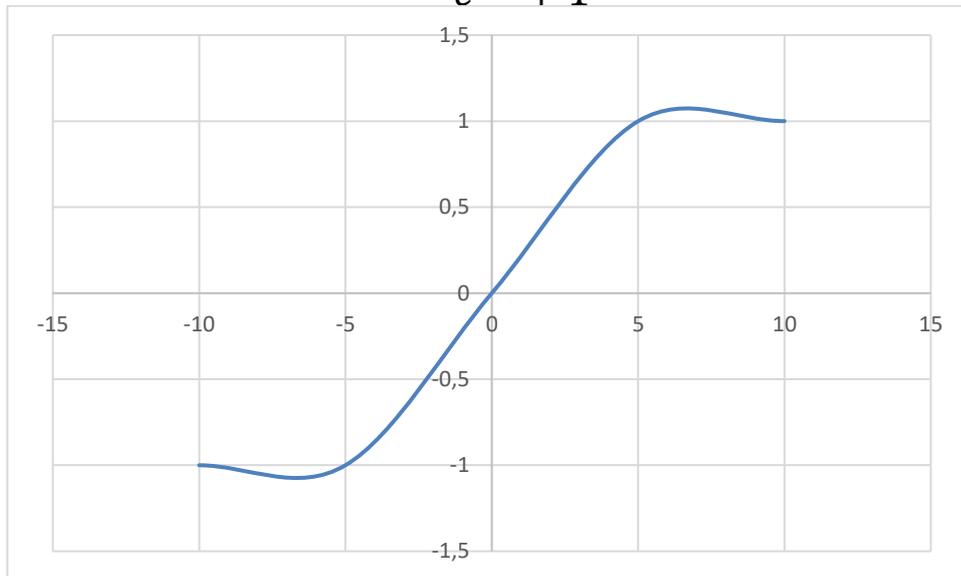


Рис. 5. Гиперболический тангенс

Линейная наиболее редко используется, а сигмоид является самой популярной. Гиперболический тангенс применяется для диапазона  $[-1;1]$  и только для него, т.к. при применении его к диапазону  $[0;1]$  уменьшается точность.

Гиперпараметр – это параметр, задаваемый вручную и, зачастую, подбираемый методом проб и ошибок.

Тренировочный сет – последовательность данных, которые использует искусственная нейронная сеть.

Итерация – это своего рода счетчик, увеличиваемый каждый раз при прохождении тренировочного сета, т.е. суммарное число пройденных сетов.

Эпоха – это гиперпараметр, означающий количество итераций. С помощью него можно обозначить насколько натренированной будет нейросеть. Соответственно, чем больше эпоха, тем лучше натренирована нейросеть. Не следует путать эпоху с итерацией и важно четко понимать, что сначала увеличивается итерация, а потом уже идет подсчет эпохи.

Ошибка в нейронных сетях является процентной величиной и должна уменьшаться в течение тренировок. Она демонстрирует расхождение между ожидаемым и полученным результатом. Существует множество различных способов вычисления ошибки, но в рамках данной статьи будут предложен к рассмотрению самый популярный метод - Mean Squared Error (MSE), формула которого заключается в подсчете ошибки за каждый сет, отняв от идеального результата, полученный. Формула MSE:

$$\frac{(i_1 - a_1)^2 + (i_2 - a_2)^2 + \dots + (i_n - a_n)^2}{n}$$

В качестве иллюстрации работы нейронной сети будет предложен пример решения задачи с применением формул функции активации (сигмоид в данном случае) и формулы нахождения ошибки.

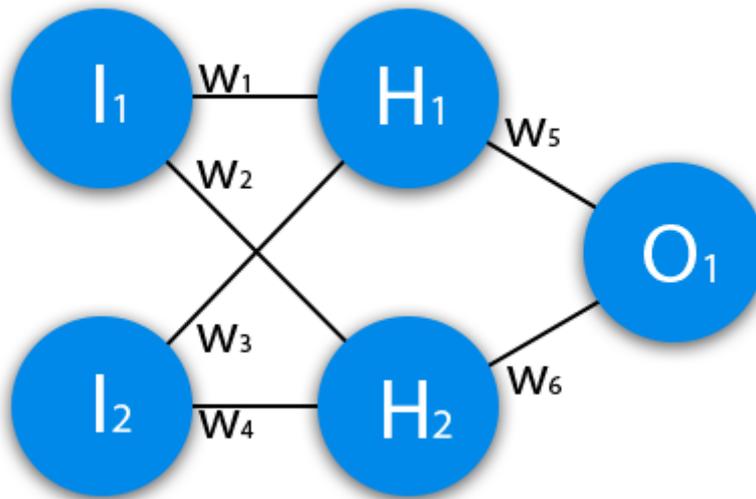


Рис.6. I – входные нейроны, H – скрытые нейроны, O – выходной нейрон, W – веса синапсов

В качестве условия изначально дано:

$$I_1 = 1$$

$$I_2 = 0$$

$$w_1 = 0.45$$

$$w_2 = 0.78$$

$$w_3 = -0.12$$

$$w_4 = 0.13$$

$$w_5 = 1.5$$

$$w_6 = -2.3$$

Решение:

$$H_{1input} = 1 * 0.45 + 0 * -0.12 = 0.45$$

$$H_{1output} = \text{sigmoid}(0.45) = 0.61$$

$$H_{2input} = 1 * 0.78 + 0 * 0.13 = 0.78$$

$$H_{2output} = \text{sigmoid}(0.78) = 0.69$$

$$O_{1input} = 0.61 * 1.5 + 0.69 * -2.3 = -0.672$$

$$O_{1output} = \text{sigmoid}(-0.672) = 0.33$$

$$O_{1ideal} = 1(0xor1) = 1$$

$$\text{Error} = \frac{(1 - 0.33)^2}{1} = 0.45$$

Результат — 0.33, ошибка — 45%

Нейрон смещения (bias) – это часто используемый определенный тип нейронов, а также имеющий ряд отличительных особенностей: вход и выход нейрона всегда равняются единице, нейрон никогда не имеет входных синапсов, синапсов между двумя нейронами смещения быть не может, таким образом, нейрон смещения может быть размещен на всех слоях, кроме выходного слоя), нейроны смещения или присутствуют в нейронной сети по одному на слое или же полностью отсутствуют.

Необходимость нейрона смещения заключается в том, чтобы иметь возможность сдвига функции активации вправо или влево.

### Метод обратного распространения.

Для того, чтобы искусственная нейронная сеть давала правильные ответы, необходимо применять методы ее обучения. Основополагающим из которых является метод обратного распространения с применением алгоритма градиентного спуска.

Для понимания этого метода следует дать определение ключевых понятий.

Так, градиентом является вектор, составленный из частных производных, который определяет крутизну склона и указывает его направление относительно какой-либо из точек на поверхности или графике.

Понятное определение градиентного спуска дается в книге «Организация и использование нейронных сетей» авторов Аксенова С.В. и Новосельцова В.Б.:

«Суть метода заключается в том, что двигаясь в направлении, противоположном градиенту функции  $\Delta W_{ij}$ , в конце концов приближаемся к минимуму, возможно локальному, функции». [5, с.15].

Локальным минимумом в свою очередь является точка в которой значение функции меньше, чем во всех точках малой окрестности [3, с. 85]. Основная проблема локальных минимумов заключается в том, что из-за них затруднительно максимально минимизировать ошибку (т.е. найти глобальный минимум). Для решение этой проблемы существует момент, который является гиперпараметром. Благодаря ему возможно преодолеть локальные минимумы, но при слишком большом моменте существует вероятность пропустить глобальный минимум. В конечном счете, все равно получится достичь глобального минимума, но при большом моменте возможно увеличение времени на достижение минимальной ошибки.

Также необходим гиперпараметр скорости обучения (learning rate). В сущности, это шаг, с которым будет проходить передвижение по графику. В его подборе важно не впасть в крайности, т.к. при очень малом шаге, обучение может, в принципе, не происходить, а при большом велика вероятность «проскочить» необходимый минимум.

После определения ключевых для метода обратного распространения понятий можно перейти к разбору самого алгоритма:

1. Происходит последовательная передача вперед (Forward pass) от входных и до выходного нейрона.
2. Вычисляется ошибка.
3. На основании найденной ошибки изменяются веса соответствующих нейронов в обратном порядке, т.е. начиная с весов выходного нейрона.

Таким образом значения нейронов будут меняться в сторону, дающую наилучший результат.

В приведенном ниже примере будут использованы методы нахождения  $\delta$  (дельта), а также стохастический метод обновления весов.

В методе  $\delta$  (дельта) применяется две формулы:

- 1)  $\delta_0 = (\text{OUT}_{\text{ideal}} - \text{OUT}_{\text{actual}}) * f'(IN)$ .
- 2)  $\delta_H = f'(IN) * \sum(w_i * \delta_i)$ .

Первая из них применяется для подсчета входных нейронов и заключается в произведении разницы идеального и полученного результата и производной функции активации от входного значения данного нейрона. Из этого также следует, что с методом обратного распространения следует применять только те функции активации, что могут быть дифференцированы.

Вторая формула применяется для скрытых нейронов и представляет из себя произведение функции активации от входного значения суммы всех исходящих весов и дельты нейрона, с которого синапс был взят.

В предыдущей задаче была вычислена ошибка (45%) и результат -0,33. При этих данных можно преступить к методу обратного распространения:

$$\begin{aligned}O_{1\text{output}} &= 0.33 \\O_{1\text{ideal}} &= 1 \\ \text{Error} &= 0.45\end{aligned}$$

$$\delta O_1 = (1 - 0.33) * ((1 - 0.33) * 0.33) = 1.48$$

На этом вычисления для нейрона окончены. После подсчета дельты необходимо сразу обновить веса всех последующих синапсов нейрона. Т.к. с  $O_1$  их нет, то следует перейти к скрытым нейронам, применяя соответствующую формулу  $\delta$ . В вычисленную  $\delta$  уже входит ошибка, поэтому вместо ошибки будет передаваться от нейрона к нейрону именно  $\delta$ .

Нахождение  $\delta$  для  $H_1$ :

$$H_{1output} = 0.61$$

$$w_5 = 1.5$$

$$\delta O_1 = 0.148$$

$$\delta H_1 = ((1 - 0.61) * 0.61) * (1.5 * 0.148) = 0.053.$$

Далее следует найти градиент по упрощенной, благодаря использованию  $\delta$ , формулы:

$$GRAD_B^A = \delta_B * OUT_A,$$

где A – это точка в начале синапса, B – точка в конце синапса.

Нахождение градиента:

$$H_{1output} = 0.61$$

$$\delta O_1 = 0.148$$

$$GRAD_{w_5} = 0.61 * 0.148 = 0.09$$

После этих вычислений имеются все данные для обновления веса  $w_5$  с помощью функции метода обратного распространения:

$$\Delta W_I = E * GRAD_W + \alpha * \Delta w_{i-1}$$

где E – скорость обучения,  $\alpha$  – момент.

Таким образом, изменение веса синапса тождественно сумме произведений коэффициента скорости обучения на градиент веса и момент умноженный на предыдущее значение веса, которое на первой итерации равно 0.

Обновление значения веса  $w_5$ :

$$E = 0.7$$

$$A = 0.3$$

$$w_5 = 1.5$$

$$GRAD_{w_5} = 0.09$$

$$\Delta w_5(i-1) = 0$$

$$\Delta w_5 = 0.7 * 0.09 + 0 * 0.3$$

$$w_5 = w_5 + \Delta w_5 = 1.563$$

Следовательно, после применения алгоритма увеличение веса составило 0,063.

Подобным же образом следует провести операцию над  $H_5$ :

$$H_{2output} = 0.69$$

$$w_6 = -2.3$$

$$\delta O_1 = 0.148$$

$$E = 0.7$$

$$A = 0.3$$

$$\Delta w_6(i-1) = 0$$

$$\delta H_2 = ((1 - 0.69) * 0.69) * (-2.3 * 0.148) = -0.07$$

$$GRAD_{w_6} = 0.69 * 0.148 = 0.1$$

$$\Delta w_6 = 0.7 * 0.1 + 0 * 0.3 = 0.07$$

$$w_6 = w_6 + \Delta w_6 = -2.2$$

Аналогичные действия над входными нейронами  $I_1$  и  $I_2$ , учитывая, что не следует находить дельты, т.к. от входных нейронов не последует далее синапсы.

Решение:

$$w_1 = 0.45, \Delta w_1(i-1) = 0$$

$$w_2 = 0.78, \Delta w_2(i-1) = 0$$

$$w_3 = -0.12, \Delta w_3(i-1) = 0$$

$$w_4 = 0.13, \Delta w_4(i-1) = 0$$

$$\delta H_1 = 0.053$$

$$\delta H_2 = -0.07$$

$$E = 0.7$$

$$A = 0.3$$

$$\text{GRAD}_{w_1} = 1 * 0.053 = 0.053$$

$$\text{GRAD}_{w_2} = 1 * -0.07 = -0.07$$

$$\text{GRAD}_{w_3} = 0 * 0.053 = 0$$

$$\text{GRAD}_{w_4} = 0 * -0.07 = 0$$

$$\Delta w_1 = 0.7 * 0.053 + 0 * 0.3 = 0.04$$

$$\Delta w_2 = 0.7 * -0.07 + 0 * 0.3 = -0.05$$

$$\Delta w_3 = 0.7 * 0 + 0 * 0.3 = 0$$

$$\Delta w_4 = 0.7 * 0 + 0 * 0.3 = 0$$

$$w_1 = w_1 + \Delta w_1 = 0.5$$

$$w_2 = w_2 + \Delta w_2 = 0.73$$

$$w_3 = w_3 + \Delta w_3 = -0.12$$

$$w_4 = w_4 + \Delta w_4 = 0.13$$

На данный момент нужно снова просчитать выходное значение и проверить изменение:

$$I_1 = 1$$

$$I_2 = 0$$

$$w_1 = 0.5$$

$$w_2 = 0.73$$

$$w_3 = -0.12$$

$$w_4 = 0.13$$

$$w_5 = 1.563$$

$$w_6 = -2.2$$

$$H_{1\text{input}} = 1 * 0.5 + 0 * -0.12 = 0.5$$

$$H_{1\text{output}} = \text{sigmoid}(0.5) = 0.62$$

$$H_{2\text{input}} = 1 * 0.73 + 0 * 0.124 = 0.73$$

$$H_{2\text{output}} = \text{sigmoid}(0.73) = 0.675$$

$$O_{1\text{input}} = 0.62 * 1.563 + 0.675 * -2.2 = -0.51$$

$$O_{1\text{output}} = \text{sigmoid}(-0.51) = 0.37$$

$$O_{1\text{ideal}} = 1(0\text{xor}1 = 1)$$

$$\text{Error} = \frac{(1 - 0.37)^2}{1} = 0.39$$

Результат — 0.37, ошибка — 39%.

Как видно из результата, в следствии данных преобразований удалось процент ошибки уменьшить на 0,04, т.е. 6%. Такие итерации нужно повторять до приближения к желаемой точности.

### **Заключение.**

В ходе данного исследования были даны ключевые определения для понимания дальнейших вычислений, а также продемонстрирован принцип работы искусственной нейронной сети на примере подсчета ошибки при методе обратного распространения.

## **БИБЛИОГРАФИЧЕСКИЙ СПИСОК**

1. Одиноких Г.А. Методы и алгоритмы биометрического распознавания человека по радужной оболочке глаза на мобильном устройстве // Теоретические основы информатики. 2019. С.2.
2. Мюллер А., Гвидо С. Введение в машинное обучение с помощью Python. Руководство для специалистов по работе с данными. Москва 2016-2017. С.13.
3. Гудфеллоу Я., Бенджио И., Курвилль А. Глубокое обучение. ДМК Москва. 2018. С. 29-85.

4. Гафаров Ф.М., Галимьянов А.Ф. Искусственные нейронные сети и их приложения. Учебное пособие. Издательство Казанского университета. 2018. С.6.
5. Аксенов С.В., В.Б. Новосельцев. Организация и использование нейронных сетей. Издательство томского политехнического университета. 2006. С.15.

## REFERENCES

1. Odinokikh G. A. Methods and algorithms of biometric recognition of a person by the iris of the eye on a mobile device // Theoretical foundations of computer science. 2019. p. 2.
2. Muller A., Guido S. Introduction to Machine Learning with Python. A guide for data scientists. Moscow 2016-2017. p. 13.
3. Goodfellow J., Benjio I., Courville A. Deep learning. DMK Moscow. 2018. p. 29-85.
4. Gafarov F. M., Galimyanov A. F. Artificial neural networks and their applications. Training manual. Kazan University Press. 2018. p. 6.
5. Aksenov S. V., V. B. Novoseltsev. Organization and use of neural networks. Tomsk Polytechnic University Press. 2006. p. 15.

## Информация об авторах

*Лазарев Михаил Павлович* - студент 1-го курса факультета «Управление на транспорте и информационные технологии», специальность «Разработка программно-информационных систем», Иркутский государственный университет путей сообщения, г. Иркутск, e-mail: [mr.lazarevskiy@inbox.ru](mailto:mr.lazarevskiy@inbox.ru).

*Яковлев Иван Владимирович* - студент 1-го курса факультета «Управление на транспорте и информационные технологии», специальность «Разработка программно-информационных систем», Иркутский государственный университет путей сообщения, г. Иркутск, e-mail: [vanbokkk@mail.ru](mailto:vanbokkk@mail.ru)

*Байкова Людмила Анатольевна* - старший преподаватель кафедры «Математика», Иркутский государственный университет путей и сообщения, г. Иркутск, e-mail: [baykova\\_la@irgups.ru](mailto:baykova_la@irgups.ru)

## Authors

*Mikhail Pavlovich Lazarev* - 1st year student of the Faculty of Transport Management and Information Technologies, specialty "Development of Software and Information Systems", Irkutsk State University of Railway Transport, Irkutsk, e-mail: [mr.lazarevskiy@inbox.ru](mailto:mr.lazarevskiy@inbox.ru).

*Ivan Vladimirovich Yakovlev* - 1st year student of the Faculty of Transport Management and Information Technologies, specialty "Development of Software and Information Systems", Irkutsk State University of Railway Transport, Irkutsk, e-mail: [vanbokkk@mail.ru](mailto:vanbokkk@mail.ru).

*Lyudmila Anatolyevna Baykova* - Senior Lecturer of the Department of Mathematics, Irkutsk State University of Railways and Communications, Irkutsk, e-mail: [baykova\\_la@irgups.ru](mailto:baykova_la@irgups.ru)

## Для цитирования

Лазарев М.П. Применение элементов теории поля в глубоком машинном обучении [Электронный ресурс] / М. П. Лазарев, И. В. Яковлев, Л. А. Байкова // Молодая наука Сибири: электрон. науч. журн. — 2021. — №12. — Режим доступа: <http://mnnv.irgups.ru/toma/121-2021>, свободный. — Загл. с экрана. — Яз. рус., англ. (дата обращения: 07.06.2021)

## For citation

Lazarev M.P., Yakovlev I.V., Baykova L.A. *Application of field theory elements in deep machine learning. Molodaya nauka Sibiri: ehlektronnyj nauchnyj zhurnal* [Young science of Siberia: electronic scientific journal], 2021, no. 12. [Accessed 07/06/21]