

**ИНФОРМАТИКА, ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА И УПРАВЛЕНИЕ**

УДК 004.032.26

DOI 10.52928/2070-1624-2026-46-1-2-9

**ЭФФЕКТИВНОЕ ОБУЧЕНИЕ НЕЙРОННЫХ СЕТЕЙ  
НА ОСНОВЕ ВЫБОРА НАЧАЛЬНОГО РЕШЕНИЯ**

*д-р техн. наук, проф. В. В. КРАСНОПРОШИН,  
канд. техн. наук В. В. МАЦКЕВИЧ  
(Белорусский государственный университет, Минск)*

*V. Krasnoproshin ORCID: <https://orcid.org/0000-0002-9463-4869>*

*V. Matskevich ORCID: <https://orcid.org/0000-0001-6964-542X>*

*Рассмотрена проблема, связанная с обучением нейронных сетей. Введено понятие неоднородности пространства оптимизации и доказан критерий его существования. Показано, что одним из факторов, влияющих на качество и скорость алгоритма обучения, является выбор начального приближения. Предложена процедура генерации начального решения, учитывающая свойство неоднородности пространства, специфику решаемой задачи и другие факторы, что позволило повысить качество обучения. Показано, что данную процедуру можно использовать для любых архитектур нейронных сетей и входных данных. Приведены результаты экспериментов по решению прикладной задачи, которые подтверждают эффективность предложенного подхода к обучению.*

**Ключевые слова:** *нейронные сети, обучение, пространство решений, неоднородность, многомерная матрица несоответствий.*

**Введение.** Нейросетевые технологии применяются для решения широкого спектра прикладных задач. Эффективность их применения напрямую зависит от качества обучения нейронных сетей. Однако при анализе процесса обучения выбору начального приближения уделяется неоправданно мало внимания. Вместе с тем, как показано в [1–4], такой выбор влияет не только на скорость обучения, но и на качество конечного решения. Можно отметить лишь отдельные результаты, которые непосредственно связаны с проблемой выбора начального решения, но они носят локальный, ограниченный характер. Они, как правило, не учитывают специфику задачи, тип архитектуры сети, свойства пространства оптимизации и др. Так, в алгоритмах градиентного типа в большей степени используют свойство сходимости. Область сходимости формируется на основе локального минимума, от которого и зависят результаты обучения. Вследствие этого начальным приближением выбирают точку пространства решений, максимально близкую к глобальному минимуму (например, локальный минимум). Именно с нее и начинают процесс обучения сети. Другие алгоритмы используют процедуры генерации начального решения на основе анализа количества весов в отдельном слое сети. Начальные приближения в этом случае строятся, как правило, на основе реализации равномерно распределенных на отрезке с центром в нуле случайных величин. Отличаются такие процедуры лишь длинами этих отрезков.

В работе исследуются свойства пространства решений оптимизационной задачи обучения. Вводится понятие его неоднородности, на основе которого предлагается процедура генерации начального приближения. Показано, что в рамках данной процедуры учитывается специфика входных данных, тип архитектуры нейронной сети и другие важные факторы. Это делает ее в некотором смысле универсальной при обучении нейронных сетей. На примере решения прикладной задачи экспериментально показано, что применение алгоритма обучения с использованием такой процедуры позволяет повысить качество результатов обучения.

**Обучение нейронных сетей.** Процесс обучения является наиболее трудоемким этапом нейросетевой технологии, который условно можно разбить на несколько частей.

Первая из них включает сбор, очистку и преобразование входных данных, разметку обучающей и тестовой выборки. Далее определяется тип архитектуры нейронной сети, которая проектируется исходя из специфики решаемой задачи. Она должна учитывать особенности входных данных и предметной области в целом. Все это подготовительная работа, которая непосредственно предшествует началу обучения. Однако полнота входных данных и выбор способа преобразования, как и типа архитектуры, существенно влияют на качество получаемого в результате обучения конечного решения.

В завершение выбирается алгоритм обучения, генерируется начальное приближение и производится непосредственная настройка нейронной сети на предметную область [5–9]. От выбора алгоритма зависит не только качество обучения, но и объем необходимых для этого вычислительных ресурсов. Это наиболее затратная с вычислительной точки зрения часть процесса.

Необходимо отметить, что обучение является типичной оптимизационной задачей с соответствующим пространством решений. Архитектура сети определяет его размерность и, возможно, некоторые дополнительные свойства, вызванные ее особенностью. Алгоритм обучения задает стратегию поиска оптимального решения, однако эффективность поиска во многом определяется свойствами структуры пространства оптимизации.

**Неоднородность пространства решений.** Рассмотрим оптимизационное пространство обучения более подробно. Известно, что архитектура любой нейронной сети задается набором слоев и связей между ними. При этом каждый слой ориентирован на выполнение одного или несколько типов преобразования, задаваемых системой параметров. Для каждого параметра определены диапазоны изменения его значений, с помощью которых можно управлять процессом обработки, то есть формально любую архитектуру нейронной сети можно задать в виде параметрического семейства  $N(X)$ , где  $X$  – система из  $m$  векторов, координаты которых задаются значениями параметров определенного типа. В пределах отдельного вектора координаты отвечают за однотипные преобразования, поэтому диапазон их допустимых значений будет также одинаковым. Из этого следует, что для каждого  $i$ -го вектора ( $i = \overline{1, m}$ ) множество допустимых значений  $D_i$  каждой координаты его параметров определяются диапазоном

$$D_i = [a_i; b_i]^{k_i},$$

где  $k_i$  – количество координат в  $i$ -м векторе,

$a_i, b_i$  – верхние и нижние границы диапазона ее значений соответственно.

Нетрудно показать, что процесс обучения нейронной сети формально можно записать в виде следующей задачи условной оптимизации. Введем обозначения:

- $N(X_1, X_2, \dots, X_m)$  – параметрическое семейство алгоритмов, реализующих преобразование информации;
- $\Omega$  – пространство параметров в виде декартова произведения множеств допустимых значений  $D_i$ , определяющих диапазоны их изменений для соответствующих векторов.

Цель задачи обучения заключается в нахождении значения параметров, доставляющих минимум некоторой функции  $F$ , фиксирующей ошибки алгоритма.

Известно, что результатом работы нейронной сети является вектор значений нейронов выходного слоя. Для различных входных данных выходные значения должны быть разными. Можно показать, что оптимальные значения параметров находятся в диапазоне достаточно узких размеров, а само оптимальное решение, в силу особенности пространства поиска, находится в некотором подпространстве малой мощности.

Действительно, при функционировании сети входные значения нейронов слоя определяются в виде суммы произведений его весов и выходных значений предыдущего слоя. Веса, как известно, фиксированы, а входные данные имеют, если задачей не оговорено, многомерное равномерное распределение. Следовательно, каждое из получаемых в результате слагаемых является равномерно распределенной на некотором отрезке случайной величиной. Из закона больших чисел известно, что по мере роста числа слагаемых их сумма сходится к нормальному распределению, которое характеризуется малой дисперсией, поэтому изначально различные значения входных данных нейронные сети преобразуют в близкие по значению взвешенные суммы. Последние используются функцией активации для построения в пространстве выходных данных разделяющей поверхности. Очевидно, что чем меньше расстояние между классами, тем точнее должна быть построена разделяющая поверхность, то есть небольшое по размеру межклассовое расстояние требует точной настройки параметров разделяющей поверхности. Это, в свою очередь, порождает неустойчивость оптимального решения. Иными словами, в окрестности оптимального решения содержатся множества точек с различными значениями целевой функции.

*Примечание.* На практике для повышения разделительной способности сети используют, как правило, веса с большим соотношением либо функции активации с большими по модулю значениями производной.

Из проведенных выше рассуждений следует вполне очевидное утверждение: *для параметров нейронной сети существует относительно небольшие отрезки, вероятность нахождения оптимальных значений в которых существенно выше, чем на оставшемся диапазоне их изменений.*

Для формализации данного факта введем следующее определение.

**Определение.** Пространство решений  $\Omega$  оптимизационной задачи обучения назовем неоднородным, если существует его разбиение на подпространства  $\Omega_1$  и  $\Omega_2$  такие, что

$$\begin{cases} \Omega = \Omega_1 \cup \Omega_2, \Omega_1 \cap \Omega_2 = \emptyset; \\ \forall \Omega_3, \Omega_4 : |\Omega_3| = |\Omega_4|, \Omega_3 \subseteq \Omega_1, \Omega_4 \subseteq \Omega_2; \\ P(x^* \in \Omega_3) \gg P(x^* \in \Omega_4), \end{cases} \quad (1)$$

где  $x^*$  – оптимальное решение.

Отношение вероятности нахождения оптимального решения в некотором подмножестве решений пространства  $\Omega$  к его мощности назовем плотностью этого подмножества.

Из введенных определений следует вполне очевидное утверждение, что если пространство  $\Omega$  обладает свойством неоднородности, то в нем существуют подмножества решений с заметно различающимися плотностями.

**Коэффициент перекоса.** Изучение свойства пространства оптимизации рассмотрим на примере решения задачи классификации. Предположим, что в качестве классификатора используется многослойный перцептрон, а входные данные разбиты на  $n$  классов. Классификацию будем производить стандартным образом: объект заносится в класс, нейрон выходного слоя которого обладает наибольшим значением.

Пусть в результате обучения построен некоторый классификатор  $N$  с набором параметров  $t = (t_1, t_2, \dots, t_m)$ , а по результатам классификации обучающей выборки получена квадратная матрица  $A(t)$  размером  $n \times n$ . Элементы матрицы  $a_{ij}(t)$  определяют количество объектов  $i$ -го класса выбранных  $j$ -м нейроном выходного слоя. Эта матрица, известная в литературе как многомерная матрица несоответствий, применяется для оценки качества обучения [10]. Сумма элементов ее главной диагонали определяют количество правильных ответов, а сумма остальных – количество неверных классификаций. Воспользуемся данными фактами и введем следующее определение.

*Определение.* Коэффициентом перекоса  $q(t)$  матрицы несоответствий  $A(t)$  назовем величину, равную отношению суммы элементов столбца матрицы с наибольшим значением к сумме всех ее элементов:

$$q(t) = \frac{\max_{j=1, \dots, n} \sum_{i=1}^n a_{ij}(t)}{\sum_{i=1}^n \sum_{j=1}^n a_{ij}(t)}. \quad (2)$$

Для задачи из  $n$  классов значение коэффициента варьируется в интервале от  $1/n$  до 1. При этом значение  $q(t) = 1$  указывает на полную «невосприимчивость» сети к данным из различных классов, а – равномерное «распределение» классов по нейронам выходного слоя, близкое к  $1/n$ .

Продемонстрируем, что с помощью данного коэффициента можно определить наличие неоднородности пространства решений. В [11] было показано, что если для некоторой сети  $N$  с набором параметров  $t = (t_1, t_2, \dots, t_m)$  коэффициент перекоса  $q(t) = 1$ , то небольшие изменения данных параметров, как правило, не изменяют значение коэффициента и, следовательно, обобщающую способность сети в рамках рассматриваемой выборки. Очевидно, это указывает на однородность пространства в окрестности рассматриваемого решения  $t$ . Неоднородность же, напротив, предполагает наличие в малой окрестности большого количества решений с различным значением обобщающей способности. С учетом указанных фактов можно сформулировать критерий неоднородности пространства решений оптимизационной задачи.

Пусть  $t = (t_1, t_2, \dots, t_m)$  значения параметров сети, полученных в результате обучения, а  $q(t)$  – соответствующее значение коэффициента перекоса. Тогда справедлива следующая теорема.

**Теорема 1.** Пространство решений  $\Omega$  неоднородно тогда и только тогда, когда в нем существует пара решений  $x$  и  $y$ , таких что

$$\max_{t \in N(x)} q(t) - \min_{t \in N(x)} q(t) \gg \max_{t \in N(y)} q(t) - \min_{t \in N(y)} q(t), |N(x)| = |N(y)| \ll |\Omega|, \quad (3)$$

где  $N(x)$  и  $N(y)$  – окрестности  $x$  и  $y$  соответственно.

**Доказательство.**

*Достаточность.* Предположим, что верно неравенство (3). Тогда существуют малые окрестности пространства решений  $N(x)$  и  $N(y)$ , в одной из которых для решения  $x$  содержится большое количество различных значений целевой функции, а в другой для решения  $y$  – малое. Это означает что вероятность нахождения оптимального решения в окрестности  $x$  значительно выше, чем в окрестности  $y$ , что и является неоднородностью пространства по определению.

*Необходимость.* Предположим, что пространство решений является неоднородным, при этом неравенство (3) не выполняется. Тогда в окрестности любого решения фиксированного размера количество возможных значений целевой функции должно быть примерно равным. Это означает, что вероятность расположения оптимального решения в окрестности любого решения является примерно одинаковой, что противоречит предположению о неоднородности пространства решений. Таким образом, утверждение доказано.

*Замечание 1.* Разница величин в неравенстве (3) может снижаться по мере роста числа классов.

*Замечание 2.* В неравенстве (3) можно использовать дисперсии коэффициента перекоса.

Воспользуемся результатами данной теоремы для разработки процедуры генерации начального приближения.

**Коэффициент перекоса и пространство решений.** В соответствии с критерием (3) для определения неоднородности пространства решений необходимо знать максимальное и минимальное значения коэффициента перекоса в малых ее областях. Аналитически определить это невозможно, так как значения коэффициента перекоса зависят не только от архитектуры нейронной сети, но и от входных данных задачи (которые нельзя задать аналитически). Количество решений в каждой малой области является несчетным. Их перебор и, следовательно, вычисление максимального и минимального значения также невозможны.

Формула (3) задана в виде строгого неравенства, причем левая его часть должна быть существенно больше правой. Для проверки соотношения таких величин, очевидно, можно использовать их грубые оценки. Действительно, погрешность оценок в таком случае существенно меньше модуля разности сравниваемых значений и, следовательно, не влияет на выполнение неравенства.

Предлагается один из возможных способов построения приближенных оценок максимального и минимального значений коэффициента перекоса.

Пусть задана нейронная сеть  $N$ , содержащая  $m$  настраиваемых параметров  $t = (t_1, t_2, \dots, t_m)$ , которыми являются веса сети. Очевидно, что в этом случае размерность пространства поиска  $\Omega$  равна  $m$ . Предположим, что в данном пространстве необходимо найти область повышенной плотности, обозначим ее через  $N(x)$ . Для нахождения такой области необходимо, очевидно, определить множество решений, которые ее образуют.

Отметим, что в критерии неоднородности (3) заданы лишь значения коэффициента перекоса и мощности окрестностей, но при этом отсутствуют какие-либо ограничения на используемую в пространстве решений метрику. Поэтому для простоты способа проверки критерия, будем полагать, что любое решение  $u$  принадлежит области  $N(x)$ , если Манхэттенское расстояние между векторами  $x$  и  $u$  не превосходит некоторого порогового значения  $\varepsilon$ .

Для построения приближенных оценок предлагается итерационная процедура, состоящая из двух этапов.

На предварительном этапе задают некоторое решение  $x = (x_1, \dots, x_m)$ , количество итераций  $k = 0, 1, \dots, h$  и полагают  $x_0 = x$ .

Затем выполняется  $k$ -я общая итерация:

- генерируют случайный вектор  $c = (c_1, \dots, c_m)$ , координатами которого являются равномерно распределенные на отрезке  $[-\varepsilon/m, +\varepsilon/m]$  случайные величины;

- формируют новое решение  $x_k = x + c$  и строят его оценки по формулам

$$\begin{cases} q_{x \min} = \min\{q_{x \min}, q(x_k)\}; \\ q_{x \max} = \max\{q_{x \max}, q(x_k)\}; \end{cases}$$

- если  $k = h$ , то останов и  $\Delta q(x) = q_{x \max} - q_{x \min}$ , в противном случае –  $k = k + 1$ .

Нетрудно заметить, что при бесконечном количестве итераций в окрестности  $N(x)$  произойдет «перебор» всех решений. Следовательно, построенная в результате приближенная оценка  $\Delta q(x)$  является состоятельной. Благодаря выбору размера отрезков, гарантируется, что  $x_k \in N(x)$ ,  $k = \overline{0, h}$ .

*Замечание.* На практике установлено, что для получения относительно точной оценки достаточно выполнить не более 10 итераций.

В результате для изучения пространства решений можно обойтись без обучения нейронных сетей: достаточно в определенной области сгенерировать несколько необученных сетей и вычислить для них на объемах обучающей выборки коэффициент перекоса.

**Генерация начального приближения.** В [11] было показано, как выбор начального решения влияет на качество конечного решения. При выборе такого приближения выдвигается дополнительное требование – его устойчивость. Иными словами, дисперсия оценки значения целевой функции должна быть наименьшей при наибольшей величине математического ожидания.

Начальное решение, очевидно, будет устойчивым, если оно расположено достаточно близко к подпространству повышенной плотности, с малым смещением за его пределы.

Начальное приближение предлагается строить с помощью алгоритма, основанного на итерационной процедуре случайного поиска оптимальной инициализации.

Для простоты изложения будем полагать, что решения  $x$  и  $y$  из неравенства (3) являются многомерными скалярами с одним значением по всем координатам ( $x = (a, a, a, \dots, a)$ ,  $y = (b, b, b, \dots, b)$ ). Такое предположение вполне допустимо, так как все рассматриваемые параметры однотипны. Предлагаемый алгоритм также состоит из двух этапов.

На предварительном этапе инициализации фиксируется архитектура обучаемой нейронной сети и производится генерация значений ее параметров. Значения всех параметров сети, кроме весов, полагают

«нейтральными», то есть не влияющими на работу сети (для смещений персептрона такими значениями, например, являются нули).

На втором (основном) этапе запускается итерационный процесс, состоящий из последовательности следующих шагов.

Шаг 1. *Выбор окрестности решения.* На первой итерации значение скаляра устанавливается равным нулю, а на всех последующих выбирается случайным образом на отрезке  $[-1, 1]$  (большие значения по модулю не имеют смысла из-за особенностей функций активации нейронов).

Шаг 2. *Оценка значения.* Для оценки выбранного решения применяется описанный выше способ. Длина отрезка для генерации определяется равной  $10^{-4}$ .

Шаг 3. *Обновление области.* При выполнении первой итерации полагают  $\Delta q_{x_{\max}} = \Delta q(x)$  и окрестность  $N(x)$  запоминается как лучшая. Окрестность  $N(x)$  обновляется тогда и только тогда, когда  $\Delta q_{x_{\max}} < \Delta q(x)$ . Новое значение  $\Delta q_{x_{\max}}$  запоминается.

Шаг 4. *Критерий останова.* Если  $\Delta q_{x_{\max}}$  за некоторое число итераций не увеличилось или оказалось наибольшим для  $x = 0$  после второй итерации, то переходим к шагу 5, в противном случае – к шагу 1.

Шаг 5. *Генерация начального решения.* Для генерации весов лучшее на текущий момент решение  $x$  смещается на небольшую величину (например, на  $10^{-4}$  в большую сторону по всем координатам) и получается решение  $x'$ . Далее генерируется одно решение из окрестности  $N(x')$ .

Шаг 6. *Улучшение начального решения.* Для решения, полученного на шаге 5, вычисляется многомерная матрица несоответствий. Затем с помощью специального алгоритма на основе динамического программирования полученное на шаге 5 решение оптимизируется [12].

Идея алгоритма основана на решении оптимизационной задачи. Целевой функцией являлось количество верных классификаций для начального решения, а параметрами оптимизации – соответствие между нейронами выходного слоя сети и классами данных. Данный алгоритм заменяет полный перебор динамическим программированием.

*Замечание 1.* Для оценки неоднородности пространства рекомендуется выполнить 5 итераций, а значение скалярных величин (для решения  $x$ ) выбирать из последовательности чисел 0; -0,5; -0,25; 0,25; 0,5.

*Замечание 2.* Результаты выполнения шага 5 не изменяют свойства пространства решений, количество настраиваемых параметров (размерность пространства решений), архитектуру нейронной сети (ограничения на значения параметров и их типы).

*Замечание 3.* Вместо разности максимального и минимального значений коэффициента перекоса можно вычислять его дисперсию.

*Замечание 4.* Если значение функции активации при нулевом значении аргумента равно 0, то область неоднородности, как правило, расположена в окрестности этого значения. Однако для нулевого значения аргумента сигмоидной функции ее значение равно 0,5. Поэтому для нее область неоднородности указанным алгоритмом может быть найдена для другого, отличного от нуля, значения скаляра.

**Оценка эффективности.** Общеизвестно, что качество получаемого при обучении решения коррелирует со значением целевой функции. Если, например, в качестве такой функции используется среднеквадратическая ошибка, то с повышением качества обучения значение ошибки, как правило, уменьшается. Поэтому очевидно, что существенное повышение обобщающей способности сети в рамках заданной обучающей выборки с большой долей вероятности приводит к значительному снижению значения целевой функции еще до момента ее обучения. Попытаемся оценить, как (или на сколько) построенное алгоритмом приближенное решение может повысить обобщающую способность сети до начала ее обучения.

Рассмотрим многомерную матрицу несоответствий. При решении задачи с  $n$  классами, как правило, применяется классификатор с  $n$  нейронами в выходном слое. Таким образом, получаем матрицу  $n \times n$ , причем сумма элементов каждой строки по условию задачи равна  $c$ . Количество данных не может быть отрицательным, следовательно, элементы матрицы – целые неотрицательные числа.

В случае классификации случайным образом получим в среднем ровно  $c$  правильных ответов, то есть  $1/n$  верных классификаций от общего количества данных.

Покажем, что сгенерированное решение гарантирует качество начального решения не хуже случайного выбора класса, то есть не менее  $1/n$ .

В соответствии с алгоритмом в каждой строке и каждом столбце матрицы выбирается ровно один элемент, поэтому всего выбирается ровно  $n$  элементов. Как было показано ранее, выбор осуществляется оптимальным образом, то есть так, чтобы максимизировать сумму выбранных элементов. Оценим теперь сумму выбранных элементов и покажем, что она всегда не меньше, чем величина  $c$ , соответствующая случайному выбору.

Доказательство проведем с использованием метода математической индукции. Вначале проверим базу индукции (для одного и двух классов).

Для одного класса это очевидно. Более того, так как все данные принадлежат одному классу и одному нейрону, получаем 100 % точность классификации.

Для двух классов имеем квадратную матрицу, состоящую из четырех элементов  $a_{11}$ ,  $a_{12}$ ,  $a_{21}$ ,  $a_{22}$ . По условию задачи сумма элементов равна  $2c$ . Стандартный выбор дает сумму  $a_{11} + a_{22}$ , то есть элементов главной диагонали. Если  $a_{11} + a_{22} \geq c$ , то факт для двух классов доказан, если же нет, то для матрицы  $2 \times 2$  возможны всего две конфигурации. Вторая оставшаяся конфигурация будет  $a_{12} + a_{21}$ . Очевидно, что

$$a_{12} + a_{21} = 2c - a_{11} - a_{22}.$$

Таким образом, получаем, что  $a_{12} + a_{21} \geq c$ , а так как в соответствии с алгоритмом выбирается максимум, то для двух классов данный факт доказан.

Теперь предположим, что данный факт верен для  $i$  классов и матрицы  $i \times i$ . Покажем, что это справедливо и для  $i + 1$  классов.

Рассмотрим матрицу размером  $(i + 1) \times (i + 1)$  и отбросим последнюю ее строку. Сумма элементов каждой строки матрицы по условию задачи равна  $c$ . По предположению, для матрицы размера  $i \times i$  существует оптимальная конфигурация, в которой сумма строк равна  $c$ , а сумма элементов больше либо равна  $c$ . Так как расширение матрицы было проведено путем добавления  $i + 1$  столбца с сохранением суммы элементов, то задача упростилась, и, следовательно, для нее существует конфигурация из  $i$  элементов суммой не меньше  $c$ . Поскольку элементы матрицы неотрицательные, то последний  $i + 1$  элемент конфигурации, выбранный из оставшегося столбца матрицы, не уменьшит общую сумму. Таким образом, было показано, что алгоритм строит решение не хуже случайного выбора, что, очевидно, является нижней для него теоретической оценкой.

*Замечание 1.* К сожалению, полученная оценка по сути является неулучшаемой. Действительно, если нейронная сеть изначально всегда дает одинаковый ответ, то оптимальная конфигурация даст ровно  $c$  правильных ответов, что и является наихудшим случаем.

*Замечание 2.* Даже если исходное начальное приближение дает более чем  $c$  правильных ответов (т. е. лучше случайного выбора), представленный алгоритм может улучшить и это начальное приближение.

Стоит отметить, что начальное приближение, сгенерированное алгоритмом, несмотря на его близость в пространстве поиска к оптимальному решению, по-прежнему является случайным и может иметь низкую обобщающую способность (гораздо хуже случайного выбора). Однако повышение качества даже до уровня случайного выбора может существенно ускорить процесс сходимости при обучении сети.

**Эксперименты.** В рамках экспериментов производилось сравнение различных способов генерации начального приближения. При этом оценивалось не только качество решения, но и среднее значение коэффициента перекоса.

Для обеспечения устойчивости и сокращения расстояния до оптимального решения начальное приближение строилось в пространстве, максимально близком к области неоднородности. В силу этого одна из целей проводимых экспериментов заключалась в исследовании зависимости качества конечного решения от близости начального решения к области неоднородности, для чего при генерации начальных значений весов рассматривались различные варианты размещения центра отрезка. Оценка близости к области неоднородности (согласно критерию неоднородности) производилась на основе вычисления для данной области разницы максимальных и минимальных значений коэффициента перекоса, а для оценки устойчивости решения вычислялось среднее значение коэффициента перекоса. В заключение исследований проводился эксперимент генерации начального решения традиционным способом. Полученные при его выполнении результаты использовались для подтверждения эффективности предложенного подхода.

Эксперименты производилось на примере ранее решенной задачи детектирования непогодных изменений поверхности Земли [13]. В качестве исходных данных использовались пары спутниковых снимков одной и той же местности, полученные в разные моменты времени. Спутниковые снимки разбивались на блоки  $20 \times 20$  пикселей, и задача решалась независимо для каждого блока. С учетом масштаба снимков блоки указанного размера вмещают наименьший возможный объект исследования.

Для упрощения условий эксперимента прикладная задача решалась с использованием простого многослойного персептрона. На вход нейронной сети в соответствии с условием задачи подавались два цветных изображения разрешением  $20 \times 20$  пикселей. Архитектура персептрона в экспериментах была фиксированной. Входной слой содержал 2400 нейронов, скрытый слой – 64 нейрона с биполярно-сигмоидной функцией активации, скрытый слой – 10 нейронов с функцией активации softmax.

Для обеспечения точности сравнения качества конечных решений время обучения было зафиксировано. Для всех испытуемых алгоритмов инициализации оно составило 4 часа. Для обучения, как и в [13], применялся тот же алгоритм на основе метода отжига [14]. Эксперименты проводились на ПК следующей конфигурации: NVIDIA RTX 3070, AMD Ryzen 7 7700, 32 GB DDR5 RAM на ОС Arch Linux.

Результаты экспериментов приведены в таблице. В первом ее столбце заданы параметры области: ее центр и размеры отрезка. В последнем указано качество конечного решения, под которым подразумевалось

относительное количество неверных классификаций. Чем меньше ошибок – тем выше качество. При этом использовалась относительная шкала. За единицу было выбрано наилучшее решение. Например, если за единицу взято наилучшее решение, которое обеспечивает точность 96 % (т. е. 4 % ошибки), то решение с точностью 92 % имеет качество, равное 0,5. В последней строке таблицы приведены результаты, полученные традиционным способом генерации начального решения.

Таблица. – Сравнение эффективности алгоритмов генерации начального приближения

Центр отрезка, длина	Среднее значение коэффициента перекося	Разница значений коэффициента перекося	Качество решения
0,0002; 0,0001	0,527	0,092	1
0,001; 0,0002	0,527	0,020	0,781
0,002; 0,001	0,527	0,035	0,658
0; 0,05	0,56	0,205	0,417

Результаты экспериментов показали, что при генерации значений параметров на отрезке с центром, отличным от нуля, начальное решение находилось за пределами области повышенной плотности. Это подтверждается неизменным средним значением коэффициента перекося. Однако по мере приближения центра к нулю разница между максимальным и минимальным значениями коэффициента возрастает. Это подтверждает факт наличия области неоднородности в оптимизационном пространстве решений в окрестности нулевых значений параметров.

Дополнительно был проведен замер разницы коэффициентов перекося для центра в нуле с длиной отрезка  $10^{-4}$ . Разница значений составила 0,200, что гораздо больше, чем для центра в точке  $2 \cdot 10^{-4}$ . Это подтверждает наличие неоднородности пространства решений. Однако генерация параметров в окрестности нуля привела к гораздо более низкому среднему качеству решения. Данный факт объясняется тем, что в области повышенной плотности решения крайне неустойчивы. Поэтому вероятность того, что начальное решение значительно хуже оптимального, очень высокая. В то же время сгенерированные за пределами данной области решения обладают устойчивостью, что обеспечивает их стабильное качество от запуска к запуску. По мере сокращения расстояния между начальным решением и областью повышенной плотности сокращается расстояние и до оптимального решения. Этим обусловлена более чем двукратная разница в качестве между первым и последним результатами в таблице при равном времени обучения. То есть начальное решение, построенное традиционным способом, обладает вдвое худшим качеством конечного решения относительно результатов, полученных с использованием предложенного алгоритма.

Отдельно стоит отметить, что последнее тестируемое решение (построенное традиционным способом) достигает качества наилучшего решения только спустя 12 часов обучения. Данный факт, очевидно, свидетельствует о том, что при равном качестве конечного решения использование предложенного алгоритма инициализации повышает также скорость обучения в целом.

Таким образом, эксперименты показали, что начальное решение, построенное предложенным алгоритмом, позволяет либо более чем вдвое повысить качество конечного результата без увеличения времени обучения, либо сократить время обучения втрое, сохранив качество решения на прежнем уровне.

**Заключение.** В работе рассмотрена проблема построения начального решения для обучения нейронных сетей. Проведено теоретическое исследование свойства пространства решений. Было введено понятие неоднородности пространства решений для задачи нейросетевой классификации и сформулирован критерий его существования. С использованием критерия был разработан алгоритм построения начального решения для обучения нейронных сетей любой архитектуры. Экспериментально на примере сравнения с традиционными алгоритмами показано, что предложенный подход позволяет повысить качество начального решения на тестируемой задаче более чем в два раза.

## ЛИТЕРАТУРА

1. Narkhede M. V., Bartakke P. P., Sutaone M. S. A review on weight initialization strategies for neural networks // *Artificial Intelligence Review* Springer. – 2022. – Vol. 55. – P. 291–322. – DOI: [10.1007/s10462-021-10033-z](https://doi.org/10.1007/s10462-021-10033-z).
2. Gradinit: Learning to initialize neural networks for stable and efficient training / C. Zhu, R. Ni, Z. Xu et al. // *Advances in Neural Information Processing Systems*. – 2021. – Vol. 34. – P. 16410–16422. – DOI: [10.48550/arXiv.2102.08098](https://doi.org/10.48550/arXiv.2102.08098).
3. de Sá G.A.G., Fontes C. H., Embiruçu M. A new method for building single feedforward neural network models for multivariate static regression problems: a combined weight initialization and constructive algorithm // *Evolutionary Intelligence*. Springer. – 2024. – Vol. 17, iss. 2. – P. 1221–1233. – DOI: [10.1007/s12065-022-00813-z](https://doi.org/10.1007/s12065-022-00813-z).
4. Ejaz S., Khurshid K. Power Spectral Density Based Weight Initialization Technique for Feed-Forward Neural Network // *2025 International Conference on Emerging Technologies in Electronics, Computing, and Communication (ICETECC) IEEE*. – 2025. – P. 1–6. – DOI: [10.1109/ICETECC65365.2025.11070241](https://doi.org/10.1109/ICETECC65365.2025.11070241).
5. Artificial neural networks training algorithm integrating invasive weed optimization with differential evolutionary model. / A. A. Movassagh, J. A. Alzubi, M. Gheisari et al. // *Journal of Ambient Intelligence and Humanized Computing*. Springer. – 2023. – Vol. 14. – P. 6017–6025. – DOI: [10.1007/s12652-020-02623-6](https://doi.org/10.1007/s12652-020-02623-6).

6. Kaveh M., Mesgari M. S. Application of Meta-Heuristic Algorithms for Training Neural Networks and Deep Learning Architectures: A Comprehensive Review // *Neural Processing Letters*. Springer. – 2023. – Vol. 55. – P. 4519–4622. – DOI: [10.1007/s11063-022-11055-6](https://doi.org/10.1007/s11063-022-11055-6).
7. Abdulkadirov R., Lyakhov P., Nagornov N. Survey of optimization algorithms in modern neural networks // *Mathematics*. MDPI. – 2023. – Vol. 11, iss. 11. – DOI: [10.3390/math11112466](https://doi.org/10.3390/math11112466).
8. Reyad M., Sarhan A. M., Arafa M. A modified Adam algorithm for deep neural network optimization // *Neural Computing and Applications*, Springer. – 2023. – Vol. 35, iss. 23. – P. 17095–17112. – DOI: [10.1007/s00521-023-08568-z](https://doi.org/10.1007/s00521-023-08568-z).
9. Naidu G., Zuva T., Sibanda E. M. A Review of Evaluation Metrics in Machine Learning Algorithms // *Artificial Intelligence Application in Networks and Systems*. CSOC 2023. Lecture Notes in Networks and Systems. Springer, Cham. – 2023. – Vol 724. – P. 15–25. – DOI: [10.1007/978-3-031-35314-7\\_2](https://doi.org/10.1007/978-3-031-35314-7_2).
10. Heydarian M., Doyle T. E., Samavi R. MLCM: Multi-Label Confusion Matrix // *IEEE Access*. – 2022. – Vol. 10. – P. 19083–19095. – DOI: [10.1109/ACCESS.2022.3151048](https://doi.org/10.1109/ACCESS.2022.3151048).
11. Мацкевич В. В. Выбор начального приближения в задачах обучения нейронных сетей // Информационные системы и технологии : материалы XI Междунар. науч. конгр. по информатике (CSIST-2025), г. Минск, 29–31 окт. 2025 г.: в 2 ч. / Бел. гос. ун-т; редкол.: С. В. Абламейко (гл. ред.) [и др.]. – Мн., 2025. – Ч. 2. – С. 196–203.
12. Мацкевич В. В. Об эффективности обучения нейронных сетей // XXV Междунар. науч.-техн. конф., посвященная 100-летию ректора ППИ Николая Петровича Сергеева и 80-летию Победы в Великой отечественной войне «Проблемы информатики в образовании, управлении, экономике и технике», г. Пенза, РФ, 31 окт. – 1 нояб. 2025 / Пензенский гос. ун-т. – Пенза, 2025 – С. 47–55.
13. Landscape's non-natural changes detection system by satellites images based on local areas / X. Zhou, Q. Bu, V. Matskevich, A. Nedzved // *Pattern Recognition and Image Analysis*. Advances in Mathematical Theory and Applications. – Vol. 34, iss. 2. – Pleidas Publishing, 2024. – P. 365–378. – DOI: [10.1134/S1054661824700159](https://doi.org/10.1134/S1054661824700159).
14. Krasnoproshin V.V., Matskevich V. V. Random search in neural networks training // *Pattern Recognition and Image Analysis*. Advances in Mathematical Theory and Applications. – Vol. 34, iss. 2. – Pleidas Publishing, 2024. – P. 309–316. – DOI: [10.1134/S105466182470010X](https://doi.org/10.1134/S105466182470010X).

Поступила 16.02.2026

## EFFECTIVE NEURAL NETWORKS TRAINING BASED ON INITIAL SOLUTION SELECTION

V. KRASNOPROSHIN, V. MATSKEVICH  
(Belarusian State University, Minsk)

*The paper deals with a problem related to neural network training. Optimization space inhomogeneity definition was introduced and its existence criteria was proven. It was shown that training algorithm speed and quality depends on, in particular, from initial solution selection. Initial solution generation procedure was proposed, which takes into account space inhomogeneity property, specifics of problem being solved and another factors, which made possible to increase training quality. It was shown that this procedure can be applied to any neural network architecture and input data. Soling applied problem experiments results are given, which approves approach efficiency to training.*

**Keywords:** neural networks, training, solution space, inhomogeneity, confusion matrix.