

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г.ЧЕРНЫШЕВСКОГО»**

Кафедра теоретических основ
компьютерной безопасности и
криптографии

РЕФЕРАТ

Радиальные нейронные сети

студента 5 курса 531 группы
направления 10.05.01—Компьютерная безопасность
факультета КНиИТ
Мызникова Сергея Анатольевич

Проверил

доцент

подпись, дата

И. И. Слеповичев

Саратов 2024

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	3
1 Теоретические основы радиальных нейросетей.....	4
1.1 Введение в понятие радиальных нейронных сетей.....	4
1.2 Математические основы.....	7
1.2.1 Задача интерполяции.....	9
1.2.2 Универсальная теорема об аппроксимации.....	12
1.3 Задача обучения радиальных нейронных сетей.....	15
1.3.1 Обучение на основе опорных векторов.....	15
1.3.2 Случайный выбор фиксированных центров.....	16
1.3.3 Выбор центров на основе самоорганизации.....	18
1.3.4 Алгоритм оптимизации “черной вдовы” (IBWO).....	21
1.4 Сравнение сетей RBF и многослойных персептронов.....	25
2 Исследования с применением RBF сетей.....	27
2.1 Сравнение RBF сетей, для задач распознавания образов с, другими методами.....	27
2.2 Использование RBF-сети с IBWO алгоритмом для задач классификации.....	29
2.3 Использование RBF-сети в медико-экологических исследованиях.....	32
ЗАКЛЮЧЕНИЕ.....	34
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	35

ВВЕДЕНИЕ

Искусственные нейронные сети, в частности радиальные нейронные сети (Radial Basis Function Networks, RBF-сети), являются мощным инструментом в решении задач, связанных с классификацией, прогнозированием и распознаванием образов.

Радиальные нейронные сети используются за счет высокой скорости обучения и способности эффективно работать с нелинейными задачами. Уникальной особенностью RBF-сетей является использование радиальных базисных функций в качестве активирующих функций, что позволяет им формировать локальные области активации в пространстве входных данных.

В данной работе рассматриваются основные принципы работы радиальных нейронных сетей, математическая модель, основные особенности, отличия данного подхода от других и приводятся примеры исследований эффективности данного вида нейронных сетей на различных наборах данных.

Целью работы является предоставление базового понимания радиальных нейронных сетей, а также изучение их возможностей в решении различных задач.

1 Теоретические основы радиальных нейросетей

1.1 Введение в понятие радиальных нейронных сетей

Нейронная сеть – это система, состоящая из многих простых вычислительных элементов, работающих параллельно, функция которых определяется структурой сети, силой взаимосвязанных связей, а вычисления производятся в самих элементах или узлах. Она содержит связанные между собой узлы (нейроны), размещенные в слоях, и адаптирует веса связей в процессе обучения для улучшения производительности.

При построении классификаторов широко используются радиальные нейронные сети. Применяемые в них радиальные базисные функции, обычно, имеют локальный характер, что облегчает выбор начальных условий для процесса обучения, обеспечивающего быстрое достижение оптимального решения. Вместе с тем, указанное обстоятельство является причиной более слабой обобщающей способности радиальных сетей по сравнению, например, с сигмоидальными. [1]

Радиально-базисные функции (RBF) — это класс функций, использующих радиальные расстояния в многомерных пространствах для аппроксимации сложных функций и алгоритмов машинного обучения. Радиальные расстояния определяются как меры расстояния между точками, основанные на радиальной симметрии и их относительном положении. [2]

Нейросетевая интерпретация сетей RBF восходит к Брумхеду и Лоу, а также к Муди и Даркену. Парк и Сандберг доказали, что сети RBF с одним скрытым слоем являются универсальными аппроксиматорами, теоретически делая более глубокие архитектуры избыточными. Брумхед и Лоу равномерно выбирали центры RBF из обучающих данных или пространства входных данных, если отсутствовали предварительные знания предметной области. Они оптимизировали уровень линейного вывода непосредственно с помощью псевдообратной матрицы. Благодаря двухуровневой архитектуре было гарантировано наличие решения задачи оптимизации. [3]

RBF используются в нейронных сетях как функции активации в скрытых слоях. Архитектура RBF-сетей включает в себя входной слой, скрытые слои с радиальными базисными функциями и выходной слой. Радиальные базисные функции позволяют обрабатывать непрерывные входные данные и определять расстояние между входными данными и центром функции. Центры нейронов и радиусы определяются в процессе обучения, и выходной слой производит результат обработки входных данных. [2]

Нейронные сети RBF используются для аппроксимации функций, прогнозирования временных рядов, классификации, распознавания образов. Помимо их высокой способности к аппроксимации эти сети обладают другими мощными характеристиками, такими как способность представлять сложные нелинейные отображения и обеспечивать быстрый и надежный механизм обучения без значительных вычислительных затрат. На рисунке 1 представлен стандартный пример радиальной нейронной сети.

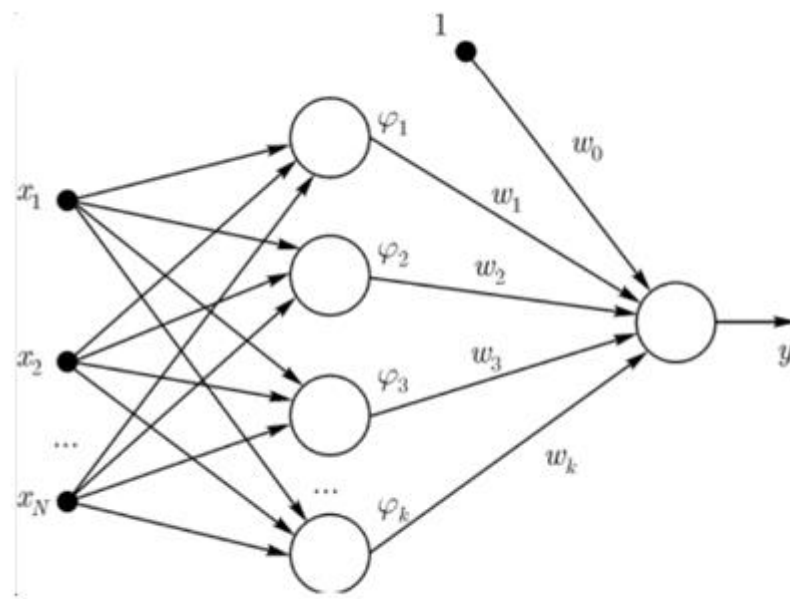


Рисунок 1 – Радиально базисная сеть [2]

В сетях с радиальными базисными функциями скрытые нейроны реализуют функции, радиально изменяющиеся вокруг выбранного центра и принимающие ненулевые значения только в окрестности этого центра.

Подобные функции называются радиальными базисными функциями, определяемые в виде:

$$\varphi_i(x) = \varphi(\|x - c_i\|),$$

где φ – радиальная функция;

x – входной вектор данных;

c – центр радиальной функции;

$\|\cdot\|$ – норма.

В таких сетях роль скрытого нейрона заключается в отображении радиального пространства вокруг одиночной заданной точки, либо вокруг группы точек, образующих кластер. Суперпозиция сигналов от скрытых нейронов, выполняемая выходным нейроном, позволяет получить отображения всей заданной области многомерного пространства.

Сигмоидальный нейрон можно представить в многомерном пространстве гиперплоскостью, которая разделяет это пространство на два класса в которых выполняется одно из двух условий:

- 1) $\sum_{i=1}^N w_{ij}x_i > 0$;
- 2) $\sum_{i=1}^N w_{ij}x_i < 0$.

Радиальный же нейрон разделяет пространство гиперсферой вокруг центральной точки и осуществляет шаровое разделение пространства. В отличие от многослойного персептрона, сеть с радиальными базисными функциями имеет как правило только один скрытый слой с числом нейронов, существенно превышающим число входов сети. Выходной нейрон осуществляет суммирование сигналов, генерируемых скрытыми нейронами.

[4]

1.2 Математические основы

Математическую основу функционирования сетей с радиальными базисными функциями (RBF-сетей) составляет теорема Ковера о разделимости образов, которая утверждает, что нелинейное преобразование сложной задачи классификации образов в пространство более высокой размерности повышает вероятность линейной разделимости образов. Рассмотрим разделимость образов. [5]

Рассмотрим семейство поверхностей, каждая из которых делит входное пространство на 2 части. Пусть X множество, состоящее из N образов (векторов) x_1, x_2, \dots, x_N , каждый из которых принадлежит одному из 2-х классов. Эта дихотомия (бинарное разбиение) точек называется разделимой по отношению к семейству поверхностей, если в этом семействе существует поверхность, которая отделяет точки класса X_1 от точек класса X_2 . Для каждого образа $x \in X$ определим вектор, состоящий из множества действительных функций $\{\varphi_i(x) | i = 1, 2, \dots, m_1\}$, вида

$$\varphi(x) = [\varphi_1(x), \varphi_2(x), \dots, \varphi_{m_1}(x)]^T.$$

Допустим m_0 это размерность векторов x , тогда векторная функция $\varphi(x)$ отображает точки m_0 -мерного векторного пространства в пространство размерности m_1 . Функции $\varphi_i(x)$ называют скрытыми, поскольку они играют роль функций активации скрытых нейронов в сетях прямого распространения. Пространство, образованное множеством скрытых функций, называют скрытым пространством или пространством признаков. [5]

Дихотомия $\{X_1, X_2\}$ множества X называется φ -разделимой, если существует m_1 -мерный вектор w , для которого можно записать

$$w^T \varphi(x) > 0, x \in X_1,$$

$$w^T \varphi(x) < 0, x \in X_2.$$

Гиперплоскость

$$w^T \varphi(x) = 0$$

описывает поверхность в скрытом пространстве. Обратный образ этой поверхности

$$x: w^T \varphi(x) = 0$$

определяет разделяющую поверхность во входном пространстве.

Рассмотрим рациональные многообразия r -го порядка во входном пространстве размерности m_0 , то есть гиперповерхности, описываемые следующим уравнением порядка r в координатах входного вектора x :

$$\sum_{0 \leq i_1 \leq i_2 \leq \dots \leq i_r \leq m_0} a_{i_1 i_2 \dots i_r} x_{i_1} x_{i_2} \dots x_{i_r} = 0,$$

где вектор x дополнен как обычно компонентом $x_0 = 1$. Для входного пространства размерности m_0 сумма содержит $\frac{(m_0-r)!}{m_0!r!}$ слагаемых, содержащих всевозможные сочетания из r множителей $x_{i_1} x_{i_2} \dots x_{i_r}$ или одночленов. [5]

В вероятностном эксперименте разделимость образов это случайное событие, зависящее от выбранной дихотомии и вероятностного распределения образов во входном пространстве. Пусть образы x_1, x_2, \dots, x_N выбираются независимо, в соответствии с вероятностным распределением, присущим входному пространству, а все возможные дихотомии $X = \{x_i\}_{i=1}^N$ равновероятны. Пусть $P(N, m_1)$ – вероятность того, что случайно выбранная дихотомия является φ -разделимой, если класс разделяющих гиперповерхностей имеет m_1 степеней свободы. Тогда

$$P(N, m_1) = \left(\frac{1}{2}\right)^{N-1} \sum_{m=0}^{m_1-1} \binom{N-1}{m}, \quad (1)$$

где $\binom{N-1}{m}$ это биномиальный коэффициент, определяемый по формуле $\binom{l}{m} = \frac{l(l-1)\dots(l-m+1)}{m!}$ для всех целых l и m . Уравнение (1) отражает сущность теоремы

Ковера о разделимости случайных образов. Само уравнение показывает вероятность того, что $(N - 1)$ подбрасываний монеты приведет к выпадению не более $(m_1 - 1)$ решек. Чем больше размерность скрытого пространства m_1 , тем ближе вероятность $P(N, m_1)$ к единице. Два основных момента вытекающих из этой теоремы. [5]

- 1) Необходимо определить скрытую функцию $\varphi_i(x), i = 1, 2, \dots, m_1$.
- 2) Более высокая размерность скрытого пространства по сравнению со входным. Эта размерность определяется числом скрытых нейронов $m - 1$.

1.2.1 Задача интерполяции

Рассмотрим сеть прямого распространения с одним входным, одним скрытым и одним выходным слоем, содержащим единственный нейрон для простоты выкладок, что, однако не снижает общности рассмотрения. Такая сеть осуществляет нелинейное отображение входного пространства в скрытое, а затем линейное отображение скрытого пространства в выходное. В целом сеть реализует отображение [5]

$$s: \mathbb{R}^{m_0} \rightarrow \mathbb{R}^1.$$

Отображение s можно рассматривать как гиперповерхность или график $\Gamma \subset \mathbb{R}^{(m_0+1)}$ изменения выходного сигнала в зависимости от входного. В общем случае эта поверхность неизвестна, а на данные обучения ещё накладывается шум. [9] На этапе обучения нейронной сети поверхность Γ оптимизируется на основании известных точек данных, представляемых сети в форме примеров, содержащих как входные, так и им соответствующие выходные данные. Фаза обобщения равносильна интерполяции на интервалах между точками, представленными в учебном наборе данных. Поверхность Γ после обучения точно проходит через все точки примеров обучения. Метод радиальных базисных функций сводится к подбору функции F , имеющий следующий вид: [5]

$$F(x) = \sum_{i=1}^N w_i \varphi(\|x - x_i\|), \quad (2)$$

где $\{\varphi(\|x - x_i\|), i = 1, 2, \dots, N\}$ – множество произвольных и обычно нелинейных функций, которые называются радиальными базисными функциями; норма используется обычно Евклидова. При этом выполняется следующее условие интерполяции:

$$F(x_i) = d_i, \quad i = 1, 2, \dots, N,$$

где d_i – желаемый выходной сигнал при подаче на вход учебного вектора x_i . Подставляя это условие в (2) можно получить систему уравнений для определения неизвестных весовых коэффициентов $\{w_i\}$:

$$\begin{bmatrix} \varphi_{11} & \varphi_{12} & \dots & \varphi_{1N} \\ \varphi_{21} & \varphi_{22} & \dots & \varphi_{2N} \\ \dots & \dots & \dots & \dots \\ \varphi_{N1} & \varphi_{N2} & \dots & \varphi_{NN} \end{bmatrix} \begin{bmatrix} w_1 \\ w_2 \\ \dots \\ w_N \end{bmatrix} = \begin{bmatrix} d_1 \\ d_2 \\ \dots \\ d_N \end{bmatrix}, \quad (3)$$

где

$$\varphi_{ij} = \varphi(\|x_j - x_i\|), \quad (i, j) = 1, 2, \dots, N.$$

Пусть

$$d = [d_1, d_2, \dots, d_N]^T,$$

$$w = [w_1, w_2, \dots, w_N]^T.$$

Векторы d и w размерности N – это вектор желаемого отклика и вектор весов соответственно. Пусть Φ – матрица размерности $N \times N$ с элементами φ_{ji} :

$$\Phi = \{\varphi_{ji} | (j, i) = 1, 2, \dots, N\}.$$

Назовем ее матрицей интерполяции. Теперь выражение (3) можно переписать в следующем виде: [5]

$$\Phi w = d.$$

Предполагая, что матрица Φ является несингулярной (следовательно, для нее существует обратная матрица Φ^{-1}), можно переписать уравнение относительно вектора весов w :

$$w = \Phi^{-1}d.$$

Для того, чтобы убедиться в несингулярности матрицы Φ , для большого класса радиальных базисных функций существует теорема Мичелли.

Теорема 1 (Теорема Мичелли).

Пусть $\{x_i\}_{i=1}^N$ – множество различных точек из \mathbb{R}^{m_0} . Тогда матрица интерполяции Φ , размерности $N \times N$ с элементами $\varphi_{ji} = \varphi(\|x_j - x_i\|)$ является несингулярной.

Теорема Мичелли охватывает большой класс радиальных базисных функций, среди которых наибольший интерес, с точки зрения сетей RBF, представляют следующие: [5]

1) Мультиквадратичная функция:

$$\varphi(r) = (r^2 + c^2)^{\frac{1}{2}}, \quad c > 0, r \in \mathbb{R};$$

2) Обратная мультиквадратичная функция:

$$\varphi(r) = \frac{1}{(r^2 + c^2)^{\frac{1}{2}}}, \quad c > 0, r \in \mathbb{R};$$

3) Функция Гаусса:

$$\varphi(r) = \exp\left(-\frac{r^2}{2\sigma^2}\right), \quad c > 0, r \in \mathbb{R}.$$

Для несингулярности матрицы Φ требуется только то, чтобы все точки $\{x_i\}_{i=1}^N$ были различными. Больше для несингулярности матрицы ничего не требуется, кроме размерности N множества примеров и размерности m_0 векторов x_i . [5]

Функции 2, 3 являются локализованными, то есть $\varphi(r) \rightarrow 0$ при $r \rightarrow 0$, а функция 1 наоборот неограниченно возрастает с ростом r , что не мешает ей тем не менее быть функцией пригодной для использования в сетях RBF, так как она является несингулярной.

Важным является тот факт, что радиальные базисные функции, неограниченно возрастающие при стремлении аргумента к бесконечности, можно использовать для аппроксимации гладких отображений с большой точностью, чем при использовании положительно-определенной матрицы. [5]

1.2.2 Универсальная теорема об аппроксимации

Пусть $G: \mathbb{R}^{m_0} \rightarrow \mathbb{R}$ - ограниченная, непрерывная и интегрируемая функция, такая, что

$$\int_{\mathbb{R}^{m_0}} G(x) dx \neq 0.$$

Пусть G_G - семейство сетей RBF, включающих функции $F: \mathbb{R}^{m_0} \rightarrow \mathbb{R}$ следующего вида:

$$F(x) = \sum_{i=1}^{m_1} w_i G\left(\frac{x - t_i}{\sigma}\right),$$

где $\sigma > 0$, $w_i \in \mathbb{R}$ и $t_i \in \mathbb{R}^{m_0}$ для $i = 1, 2, \dots, m_1$. Тогда выполняется следующая теорема об универсальной аппроксимации для сетей RBF.

Теорема 2 (Теорема об универсальной аппроксимации для сетей RBF).

Для любой непрерывной функции $f(x)$ найдется сеть RBF с множеством центров $\{t_i\}_{i=1}^{m_1}$ и общей шириной $\sigma > 0$, такая, что функция $F(x)$, реализуемая сетью, будет близка к $f(x)$ по норме L_p , $p \in [1, \infty]$. [5]

Данная теорема является более строгой, чем это необходимо для сетей RBF, так как по ней ядро $G: \mathbb{R}^{m_0} \rightarrow \mathbb{R}$ не обязательно должно удовлетворять условию радиальной симметрии. Она подводит теоретический базис под

построение нейронных сетей на основе радиальных базисных функций с целью их практического применения.

Необходимо рассмотреть также вопрос порядка аппроксимации, так как внутренняя сложность класса аппроксимирующих функций экспоненциально возрастает с увеличением отношения:

$$\frac{m_0}{s},$$

где m_0 – размерность входного сигнала;

s – индекс гладкости, определяемый как количество ограничений, накладываемых на аппроксимирующие функции этого конкретного класса.

Существует проблема, сформулированная Беллманом, “проклятие размерности”, означающая, что независимо от используемого метода аппроксимации при постоянном индексе гладкости s количество параметров, необходимых для достижения требуемого уровня точности функции аппроксимации, экспоненциально возрастает в зависимости от размерности входного пространства m_0 . Единственный способ добиться независимости скорости сходимости от размерности входного сигнала m_0 и, таким образом, избежать проблемы – увеличить индекс гладкости пропорционально увеличению количества параметров в аппроксимирующей функции, что ведет к увеличению ее сложности. [5]

Таблица 1 – Два способа аппроксимации и соответствующие функциональные пространства с одинаковой скоростью сходимости $O(\frac{1}{\sqrt{m_1}})$, где m_1 – размерность скрытого пространства. [5]

Функциональное пространство	Норма	Техника аппроксимации
$\int_{\mathbb{R}^{m_0}} \ s\ \tilde{F}(s) ds < \infty$, где $F(s)$ – многомерное преобразование Фурье для функции аппроксимации $F(x)$.	$L_2(\Omega)$	а) Многослойный перцептрон: $F(x) = \sum_{i=1}^{m_1} a_i \varphi(w_i^T x + b_i)$, где $\varphi(\cdot)$ – сигмоидальная функция активации.
Пространство функций Соболева, произведение которых до порядка $2m > m_0$ являются интегрируемыми.	$L_2(\mathbb{R}^2)$	б) RBF-сети: $F(x) = \sum_{i=1}^{m_1} a_i \exp\left(-\frac{\ x - t_i\ ^2}{2\sigma^2}\right).$

В таблице 1 приведены ограничения на функциональные пространства, которым должны удовлетворять рассматриваемые способы аппроксимации – многослойный перцептрон и RBF-сети, для того чтобы скорость сходимости не зависела от размерности входного пространства m_0 . Эти условия для различных методов аппроксимации отличаются. В случае RBF используется пространство Соболева функций, производные которых до порядка $2m > m_0$ являются интегрируемыми. То есть, чтобы скорость сходимости не зависела от роста размерности, необходимо с ростом размерности входного пространства m_0 увеличивать количество производных функции аппроксимации, которые должны быть интегрируемыми. Из таблицы можно сделать вывод:

Пространства функций аппроксимации, достижимые многослойным перцептроном и сетями RBF, становятся все более ограниченными по мере роста размерности входного пространства m_0 . Из этого следует, что проблему “проклятия размерности” нельзя преодолеть. [5]

1.3 Задача обучения радиальных нейронных сетей

Обучение RBF сетей обычно проходит в 2 этапа:

- 1) Настройка параметров слоя RBF, включая центры RBF $c_j \in \mathbb{R}^d, j = 1, \dots, K$, и параметры масштабирования, заданные с помощью скаляров, векторов или матриц $R_j \in \mathbb{R}, \in \mathbb{R}^d$ или \mathbb{R}^{d^2} соответственно. d – размерность входного пространства.
- 2) Вычисление выходных весов $w_j \in \mathbb{R}^L$ для $j = 1, \dots, L$ сети. L – количество классов. [6]

1.3.1 Обучение на основе опорных векторов

Метод опорных векторов (SVM) изначально был разработан для классификации данных линейно разделимых наборов. В этом случае тренировочный набор, состоящий из M примеров (x^μ, y^μ) , где $x^\mu \in \mathbb{R}^d$ – точка данных, $y^\mu \in \{-1, 1\}$ – метка класса, можно разделить на две группы с помощью разделяющей гиперплоскости. Такая гиперплоскость определяется весовым вектором $w \in \mathbb{R}^d$ и смещением $\theta \in \mathbb{R}$, которые удовлетворяют следующему ограничению:

$$y^\mu[(w \cdot x^\mu) + \theta] \geq 1, \quad \mu = 1, \dots, M.$$

Расстояние между разделяющей гиперплоскостью и ближайшими точками тренировочного набора называется зазором. Чем больше зазор, тем выше способность гиперплоскости к обобщению. Оптимальная гиперплоскость с максимальным зазором уникальна и может быть выражена в виде линейной комбинации тех точек, которые находятся непосредственно на границе зазора. Эти точки называются опорными векторами. Эта разделяющая гиперплоскость с максимальным зазором имеет вид: [6]

$$H(x) = \sum_{\mu=1}^M \alpha_\mu^* y^\mu \langle x, x^\mu \rangle + \alpha_0^*,$$

где $\alpha_1^*, \dots, \alpha_M^*$ – решения оптимизационной задачи.

Оптимальные коэффициенты α_μ находятся путем решения следующей задачи оптимизации:

$$Q(\alpha) = \sum_{\mu=1}^M \alpha_\mu - \frac{1}{2} \sum_{\mu=1}^M \sum_{v=1}^M \alpha_\mu \alpha_v y^\mu y^v \langle x^\mu, x^v \rangle,$$

с учетом ограничений $\alpha_\mu \geq 0$ для всех $\mu = 1, \dots, M$,

$$\sum_{\mu=1}^M \alpha_\mu y^\mu = 0.$$

Вектор обучающего набора x^μ является опорным вектором, если соответствующий коэффициент $\alpha_\mu^* > 0$. Тогда весовой вектор w имеет вид:

$$w = \sum_{\mu=1}^M \alpha_\mu y^\mu x^\mu = \sum_{\mu, \alpha_\mu > 0} \alpha_\mu y^\mu x^\mu,$$

и смещение α_0^* определяется через один из опорных векторов (x^s, y^s) как: [6]

$$\alpha_0^* = y^s - \langle w, x^s \rangle.$$

1.3.2 Случайный выбор фиксированных центров

Предлагается использование фиксированных радиальных базисных функций, определяющих функции активации скрытых элементов. Размещение центров может быть выбрано случайным образом из множества данных примеров. Для реализации радиальных базисных функций можно использовать изотропные функции Гаусса, стандартное отклонение которых является фиксированным относительно разброса центров. В частности, радиальные базисные функции с центром в точке t_i определяются выражением: [5]

$$G(\|x - t_i\|^2) = \exp\left(\frac{m_1}{d_{max}^2} \|x - t_i\|^2\right), \quad i = 1, 2, \dots, m_1,$$

где m_1 – количество центров;

d_{max} – максимальное расстояние между выбранными центрами.

В результате стандартное отклонение всех радиальных базисных функций Гаусса будет фиксированным:

$$\sigma = \frac{d_{max}}{\sqrt{2m_1}}.$$

Эта формула гарантирует, что отдельные радиальные базисные функции не будут слишком гладкими или слишком остроконечными.

При использовании этого подхода единственными параметрами, которые настраиваются в процессе обучения сети, являются синаптические веса ее выходного слоя. Их проще всего настроить с помощью метода псевдообращения. В частности: [5]

$$w = G^+ d,$$

где d – вектор желаемого отклика для множества примеров;

G^+ – матрица, являющаяся псевдообратной матрице G .

Сама матрица G определяется следующим образом:

$$G = \{g_{ji}\},$$

где

$$g_{ji} = \exp\left(-\frac{m_1}{d^2} \|x_j - t_i\|^2\right), \quad j = 1, 2, \dots, N, \quad i = 1, 2, \dots, m_1,$$

где x_j – j -й входной вектор множества примеров обучения.

Основой всех алгоритмов вычисления псевдообратных матриц является сингулярная декомпозиция.

Если G – действительная матрица размерности $N \times M$, то существуют ортогональные матрицы

$$U = \{u_1, u_2, \dots, u_N\}$$

и

$$V = \{v_1, v_2, \dots, v_M\},$$

такие, что

$$U^T G V = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_K), K = \min(N, M),$$

где

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_K > 0.$$

Векторы-столбцы матрицы U называют левыми сингулярными векторами матрицы G , а векторы-столбцы матрицы V – правыми сингулярными векторами. Числа $\sigma_1, \sigma_2, \dots, \sigma_K$ называют сингулярными значениями матрицы G . Псевдообратная матрице G , размерности $M \times N$ определяется следующим образом: [5]

$$G^+ = V \Sigma^+ U^T,$$

где Σ^+ – матрица размерности $N \times N$, выраженная в терминах сингулярных значений матрицы G :

$$\Sigma^+ = \text{diag}\left(\frac{1}{\sigma_1}, \frac{1}{\sigma_2}, \dots, \frac{1}{\sigma_K}, 0, \dots, 0\right).$$

1.3.3 Выбор центров на основе самоорганизации

Предыдущий метод для эффективной работы требует большое множество примеров. Одним из способов обхода этой проблемы является использование гибридного процесса обучения, состоящего из 2 этапов:

- 1) Этап обучения на основе самоорганизации. Его целью является оценка подходящих положений центров радиальных базисных функций скрытого слоя.
- 2) Этап обучения с учителем. На этом этапе создание сети завершается оценкой линейных весов выходного слоя.

Для процесса обучения на основе самоорганизации требуется разработать алгоритм кластеризации, разбивающий заданное множество точек данных на две подгруппы, каждая из которых должна быть максимально однородной. Один из таких алгоритмов называется алгоритм кластеризации называется алгоритм кластеризации k -средних. Согласно этому алгоритму, центры радиальных базисных функций размещаются только в тех областях входного пространства X , в которых имеются информативные данные. Пусть m_1 – количество радиальных базисных функций. Определение подходящего значения для m_1 требует проведения некоторых экспериментов. Пусть $\{t_k(n)\}_{k=1}^{m_1}$ – центры радиальных базисных функций на n -й итерации алгоритма. Тогда алгоритм кластеризации по k -средним можно описать следующим образом. [5]

- 1) Инициализация. Выбираем случайные значения для исходных центров $t_k(0)$. Единственным требованием к их выбору на данном шаге является различие всех начальных значений. При этом значения Евклидовой нормы по возможности должны быть небольшими;
- 2) Выборка. Выбираем вектор x из входного пространства X с определенной вероятностью. Этот вектор будет входным для алгоритма на итерации n ;
- 3) Проверка подобия. Обозначим $k(x)$ индекс наиболее подходящего центра для данного вектора x . Находим $k(x)$ на итерации n , используя критерий минимального Евклидова расстояния:

$$k(x) = \arg \min_k \|x(n) - t_k(n)\|, \quad k = 1, 2, \dots, m_1,$$

где $t_k(n)$ – центр k -й радиально базисной функции на итерации n ;

- 4) Корректировка. Корректируем центры радиальных базисных функций, используя правило:

$$t_k(n+1) = \begin{cases} t_k(n) + \eta[x(n) - t_k(n)], & k = k(x), \\ t_k(n) - \text{иначе}, \end{cases}$$

где η – параметр скорости обучения из диапазона $0 < \eta < 1$;

- 5) Продолжение. Увеличиваем на единицу значение n и возвращаемся к шагу 2, продолжая процедуру до тех пор, пока положение центров t_k существенно изменяется. [5]

Ограничением алгоритма является нахождение только одного локального оптимального решения, зависящего от исходного выбора центров кластера. Следовательно, вычисления могут быть напрасными: отдельные центры изначально попадут в те области входного пространства X , где количество точек данных невелико и откуда не будет шанса переместиться в области, требующие большого количества центров. В результате получится неоправданно большая сеть. Для обхода этого недостатка был предложен улучшенный алгоритм кластеризации по k -средним, который основан на понятии взвешенной переменной меры принадлежности кластеру, обеспечивающем сходимость алгоритма к оптимальной или квазиоптимальной конфигурации, независимо от исходного положения центров. [5]

Определив отдельные центры гауссовых радиальных базисных функций и их общий вес с помощью алгоритма кластеризации можно перейти к оценке весов выходного слоя. Простейшим методом оценки является алгоритм LMS (least-mean-square). Вектор выходного сигнала, сгенерированного скрытыми узлами, является входным вектором алгоритма LMS.

Алгоритм минимизации среднеквадратической ошибки основан на использовании дискретных значений функции стоимости:

$$E(w) = \frac{1}{2} e^2(n),$$

где $e(n)$ – сигнал ошибки, измеренный в момент времени n .

Дифференцируя $E(w)$ по вектору весов w , получим:

$$\frac{\partial E(w)}{\partial w} = e(n) \frac{\partial e(n)}{\partial w}.$$

Для работы с линейным нейроном сигнал ошибки можно записать в виде:

$$e(n) = d(n) - x^T(n)w(n).$$

Следовательно,

$$\frac{\partial e(n)}{\partial w(n)} = -x(n),$$

$$\frac{\partial E(w)}{\partial w(n)} = -x(n)e(n).$$

Используя полученный результат, можно оценить вектор градиента:

$$\hat{g}(n) = -x(n)e(n).$$

Используя эту оценку, можно сформулировать алгоритм минимизации среднеквадратической ошибки в следующем виде:

$$\hat{w}(n+1) = \hat{w}(n) + \eta x(n)e(n),$$

где η – параметр скорости обучения. [5]

1.3.4 Алгоритм оптимизации “черной вдовы” (IBWO)

Рассматривается трехслойная RBFNN, состоящая из входного слоя, скрытого слоя и выходного слоя. В качестве функции активации используется функция ядра Гаусса:

$$h_j = e^{-\frac{\|x - c_j\|^2}{2\sigma_j^2}}.$$

Связь между скрытым слоем и выходным слоем описывается уравнением:

$$y_k = \sum_{j=1}^h w_{jk} h_j .$$

где h_j – j -й нейрон в скрытом слое;

y_k – k -й нейрон в выходном слое;

x – входной вектор;

w_{jk} – выходной вес между j -м скрытым слоем и k -м выходным нейроном;

c_j – центральный вектор j -го скрытого нейрона;

σ_j – ширина j -го скрытого нейрона;

$\|\cdot\|$ – евклидово расстояние между входным вектором и центральным вектором. [7]

Подобно другим алгоритмам роевого интеллекта, алгоритм BWO ищет оптимальное решение на основе репродуктивного поведения пауков черной вдовы (скрещивание, каннибализм и мутация), что имеет преимущества быстрой скорости сходимости и высокой точности оптимизации. Алгоритм имеет следующие шаги:

- 1) Алгоритм инициализирует случайную популяцию пауков. Каждая особь представлена вектором-кандидатом решения алгоритма, то есть $[b_1, b_2, \dots, b_k]$;
- 2) Функция пригодности решаемой задачи нужна для расчета качества решения. Поскольку сильное потомство выживает, кандидаты на решения с меньшими значениями функции пригодности сохраняются. Лучшее решение сохраняется, а худшее удаляется;

- 3) Поведение скрещивания – это процесс, при котором оба родителя производят потомство в соответствии с коэффициентом скрещивания. Математическая модель будет выглядеть так:

$$\begin{cases} y_1 = \alpha * b_1 + (1 - \alpha) * b_2 \\ y_2 = \alpha * b_2 + (1 - \alpha) * b_1 \end{cases}$$

где каждый элемент массива α является случайным значением в диапазоне $[0, 1]$;

b_1, b_2 – являются родителями;

y_1, y_2 – являются потомками;

* - операция поэлементного умножения векторов. [7]

Процесс скрещивания повторяется $\frac{k}{2}$ раз. Каннибализм относится к тому факту, что у обоих родителей сильные самки паука поедают слабых самцов. Среди потомства сильные потомки съедают слабых потомков. Мутационное поведение – это процесс, посредством которого исходная особь производит новую особь в соответствии со скоростью мутации. Процесс мутации показан на рисунке 2, $[b_1, \dots, b_i, \dots, b_j, \dots, b_k]$ мутирует в $[b_1, \dots, b_j, \dots, b_i, \dots, b_k]$. Этот шаг заключается в поиске других особей по всему пространству поиска, улучшении разнообразия популяции и обновлении возможных решений; [7]

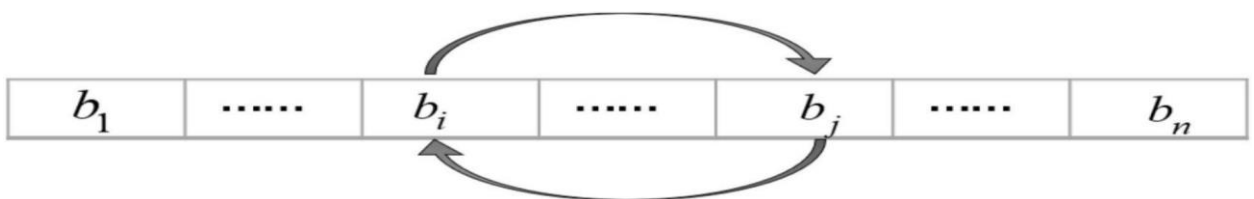


Рисунок 2 – Мутационное поведение особей пауков [7]

- 4) Циклически повторяются шаги 2 и 3. Возможные решения непрерывно обновляются, а значения функции пригодности сравниваются до тех пор, пока не будет достигнуто максимальное количество итераций или не будет достигнуто оптимальное значение. [7]

Алгоритм BWO имеет тенденцию страдать от преждевременной сходимости к точке. Можно задать массив β , вместо того, чтобы использовать случайное значение, и он будет изменяться с течением времени. Уравнение для массива β алгоритма IBWO будет иметь вид:

$$\beta_i = 8 \times \left(\frac{2}{iter} \right)^{0.3}, \quad \beta = [\beta_1, \beta_2, \dots, \beta_k].$$

Таким образом, уравнение обновления для процесса скрещивания будет иметь вид:

$$\begin{cases} y_1 = \alpha * b_1 + (1 - \alpha) * b_2 \\ y_2 = \beta * b_2 + (1 - \beta) * b_1' \end{cases}$$

где b_1, b_2 – родители;

y_1, y_2 – потомки;

$iter$ – текущее число итераций;

8 – константа массива β , определенная экспериментальным путем, которая усиливает глобальную поисковую способность алгоритма. [7]

Алгоритм IBWO одновременно ищет центральный вектор, ширину и выходной вектор RBFNN. Алгоритм IBWO и RBFNN связаны векторами-кандидатами решения алгоритма IBWO, показанными на рисунке 3.

$C_1^T \quad \sigma_1 \quad \omega_1^T$	$C_2^T \quad \sigma_2 \quad \omega_2^T$	$\dots\dots\dots$	$C_h^T \quad \sigma_h \quad \omega_h^T$
---	---	-------------------	---

Рисунок 3 – Структура каждой черной вдовы [7]

Где C_j, σ_j, w_j – центральный вектор, ширина и выходной вектор j -го скрытого нейрона для каждой особи черной вдовы соответственно. Количество нейронов скрытого слоя в модели IBWO-RBF определяет размер сети. Количество нейронов входного слоя IBWO-RFB определяется размерностью данных. Количество нейронов в выходном слое определяется количеством категорий задач классификации.

Для задач классификации и регрессии среднеквадратическая ошибка (RSME) используется как функция пригодности для оптимизации. К примеру, для задачи классификации конкретное уравнение расчета выражается как:

$$RMSE = \sqrt{\frac{1}{N} \sum_{k=1}^N (y_k - f(x_k))^2}.$$

Ожидается, что RMSE будет как можно ближе к 0. Следовательно, цель процесса поиска – получить оптимальное решение, которое минимизирует значение функции пригодности. [7]

1.4 Сравнение сетей RBF и многослойных персептронов

Сети на основе радиальных базисных функций (RBF) и многослойный персептрон (MLP) являются примерами нелинейных многослойных сетей прямого распространения. И те и другие являются универсальными аппроксиматорами. Таким образом, неудивительно, что всегда существует сеть RBF, способная имитировать многослойный персептрон (и наоборот). Однако эти два типа сетей отличаются по некоторым важным аспектам. [5]

- 1) RBF-сети в своей базовой форме имеют один скрытый слой, а многослойный персептрон может иметь большее количество скрытых слоев. Поскольку RBF-сети моделируют произвольную не линейную функцию с помощью одного промежуточного слоя, они избавляют от необходимости решать вопрос о числе слоев НС;
- 2) В многослойном персептроне для скрытых слоев и выходного слоя используется одна и та же модель нейрона, в RBF-сети скрытого слоя могут в корне отличаться от нейронов выходного слоя, и служить разным целям;
- 3) Скрытый слой с RBF-сети является нелинейным, а выходной – линейным. В многослойном персептроне, который используется в качестве классификатора, скрытые и выходной слой являются

нелинейными. При использовании многослойного персептрона для решения задач регрессии в качестве выходных нейронов обычно выбираются линейные нейроны. Параметры линейной комбинации в выходном слое RBF-сети можно оптимизировать с применением известных методов линейной оптимизации, эти методы не испытывают трудности с локальными минимумами, которые мешают при обучении многослойных персептронов с применением алгоритма обратного распространения ошибки;

- 4) Аргумент каждого скрытого нейрона RBF-сети представляет собой евклидову норму (расстояние) между входным вектором и центром радиальной функции. В то же время аргумент функции активации каждого скрытого нейрона многослойного персептрона – это скалярное произведение входного вектора и вектора синаптических весов данного нейрона;
- 5) Многослойный персептрон обеспечивает глобальную аппроксимацию нелинейного отображения. С другой стороны, сеть RBF с помощью экспоненциально уменьшающихся локализованных нелинейностей создает локальную аппроксимацию нелинейного отображения;
- 6) Для аппроксимации нелинейного отображения с помощью многослойного персептрона может потребоваться меньшее число параметров, чем для сети RBF при одинаковой точности вычислений.

[5]

2 Исследования с применением RBF сетей

2.1 Сравнение RBF сетей, для задач распознавания образов с, другими методами

В исследовании, проведенным Фабианом Вюрцбергером и Фридрихельмом Швенкером, модель RBF сети использовалась для классификации изображений и распознавания речевых эмоций. Также в сравнение ставились другие подходы. В качестве наборов данных использовались следующие: [3]

- 1) MNIST — это набор данных изображений в оттенках серого, часто используемый для оценки моделей машинного обучения. Набор данных состоит из 28×28 пиксельных изображений рукописных цифр от нуля до девяти, дающие задачу классификации по 10 классов. Всего имеется 60 000 обучающих образцов и 10 000 тестовых образцов;
- 2) CIFAR10 состоит из RGB-изображений десяти различных классов, содержащих различные объекты и животных. Каждое изображение имеет 32×32 пикселей, что дает 3072 признака на изображение. Весь набор данных имеет такой же размер, как и набор данных MNIST, содержащий 50 000 обучающих образцов и 10 000 тестовых образцов.

Таблица 2. Сравнение точности тестов на MNIST [3]

Модель	Тест точности (%)	Число параметров
DeepRBF	99.50 ± 0.07	229,160
Stochastic Pooling	99.53	-
DeepRBF Ensemble	99.67	1,145,800
CNN	99.87	1,514,187

Таблица 3. Сравнение точности тестов на CIFAR10 [3]

Модель	Тест точности (%)	Число параметров
DeepRBF	80.72 ± 0.63	268,600
Stochastic Pooling	84.87	-
DeepRBF Ensemble	85.01	1,343,000
ResNet110	93.57	17,000,000
EfficientNetV2	99.10	121,000,000
ViT-H/14	99.50 ± 0.06	632,000,000

Для сравнения использовались следующие модели:

- 1) DeepRBF – это нейронная сеть с радиально-базисной функцией;
- 2) Stochastic Pooling – это метод агрегации в свёрточных нейронных сетях. Вместо стандартного max или average pooling, он выбирает значения на основе вероятностного распределения, что добавляет элемент случайности и помогает модели лучше обобщать данные;
- 3) DeepRBF Ensemble – это комбинация нескольких моделей RBF;
- 4) CNN – свёрточные нейронные сети;
- 5) ResNet – это архитектура глубоких нейронных сетей. ResNet110 содержит 110 слоёв. Её ключевая особенность – остаточные соединения, которые помогают бороться с затуханием градиента в очень глубоких сетях;
- 6) EfficientNetV2 – это улучшенная версия EfficientNet, предназначенная для большей скорости обучения и эффективности использования ресурсов;
- 7) ViT-H/14 – это трансформер для обработки изображений. [3]

Как можно увидеть в таблице 2, нейронная сеть RBF достигает довольно неплохого результата 95.5% в распознавании образов на тестовом наборе MNIST. Но вот для набора CIFAR10 из таблицы 3, точность довольно средняя и составляет всего 80.72%, даже после использования комбинации из 5 сетей RBF точность составила 85.01%, что сильно отстаёт от других моделей. [3]

2.2 Использование RBF-сети с IBWO алгоритмом для задач классификации

Следующее исследование проводилось для сети RBF, основанной на улучшенном алгоритме оптимизации “черной вдовы” (IBWO). Для тестирования использовались наборы данных, представленные на таблице 4. [7]

Таблица 4 – Описание наборов классификационных данных [7]

Набор данных	Количество признаков	Количество обучающих выборок	Количество тестовых выборок
Australian	14	455	235
Blood	4	493	255
Breast	8	461	238
Diabetes	8	506	262
Diagnosis I	6	79	41
German	24	660	340
Diagnosis II	6	79	41
Hepatitis	10	102	53
Liver	6	227	118
Vertebral	6	204	106

В качестве конкурентов тестируемому алгоритму использовались следующие:

- 1) ABWO – адаптивный алгоритм оптимизации “черной вдовы”;
- 2) BWO – алгоритм оптимизации “черной вдовы”;
- 3) BBO – алгоритм оптимизации на основе биогеографии. Этот метод моделирует миграцию видов между регионами и используется для оптимизации;
- 4) GA – генетический алгоритм. Этот эволюционный метод основан на принципах естественного отбора и генетики для поиска решений;

- 5) PSO – алгоритм оптимизации роя частиц. Основан на коллективном поведении частиц, которые двигаются в поиске оптимального решения;
- 6) ACO – алгоритм оптимизации муравьиных колоний. Имитирует поведение муравьёв при поиске кратчайшего пути к источнику пищи.

В таблице 5 представлены результаты работы алгоритмов на наборе классификационных данных. [7]

Таблица 5 – Средняя степень классификации для алгоритмов [7]

Набор данных	IBWO	ABWO	BWO	BBO	GA	PSO	ACO
Australian	81.75%	81.02%	76.12%	77.45%	77.22%	76.90%	76.39%
Blood	98.32%	97.90%	88.28%	97.86%	96.89%	93.15%	68.53%
Breast	90.00%	89.81%	81.13%	84.72%	84.72%	83.58%	66.42%
Diabetes	77.94%	77.56%	69.81%	71.60%	70.00%	67.52%	60.08%
Diagnosis I	87.23%	84.85%	70.00%	78.40%	77.08%	74.06%	64.53%
German	100.0%	100.0%	100.0%	100.0%	100.0%	86.34%	57.32%
Diagnosis II	100.0%	100.0%	91.95%	100.0%	100.0%	88.54%	57.32%
Hepatitis	72.37%	70.17%	64.41%	69.15%	64.66%	58.98%	52.20%
Liver	74.08%	72.97%	71.18%	71.91%	68.09%	56.38%	55.91%
Vertebral	88.13%	85.60%	79.10%	85.32%	84.55%	71.57%	58.85%

Также на рисунках 5 и 6 были представлены графики с результатами аппроксимации и идентификации алгоритма IBWO.

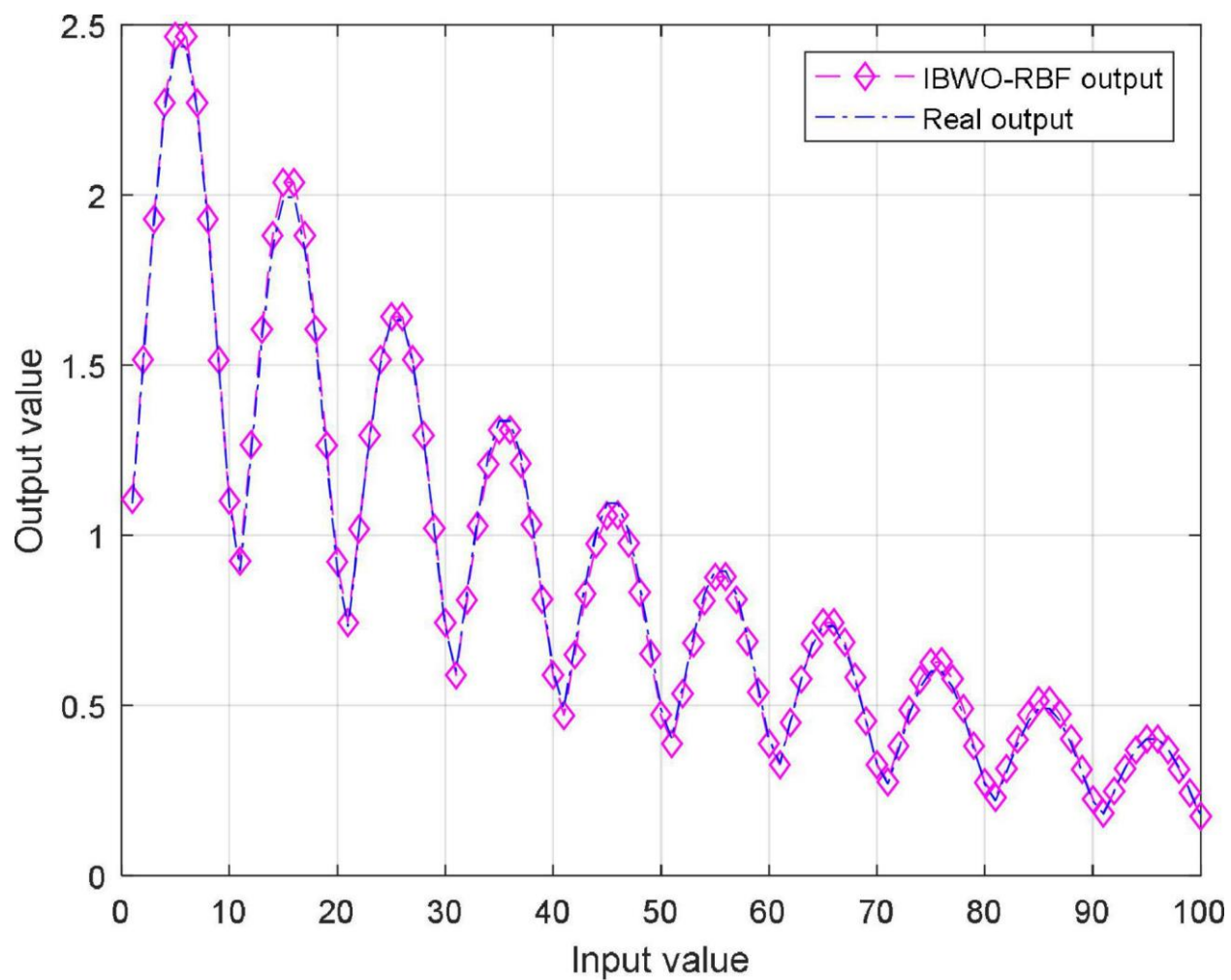


Рисунок 5 – Результаты аппроксимации алгоритма IBWO [7]

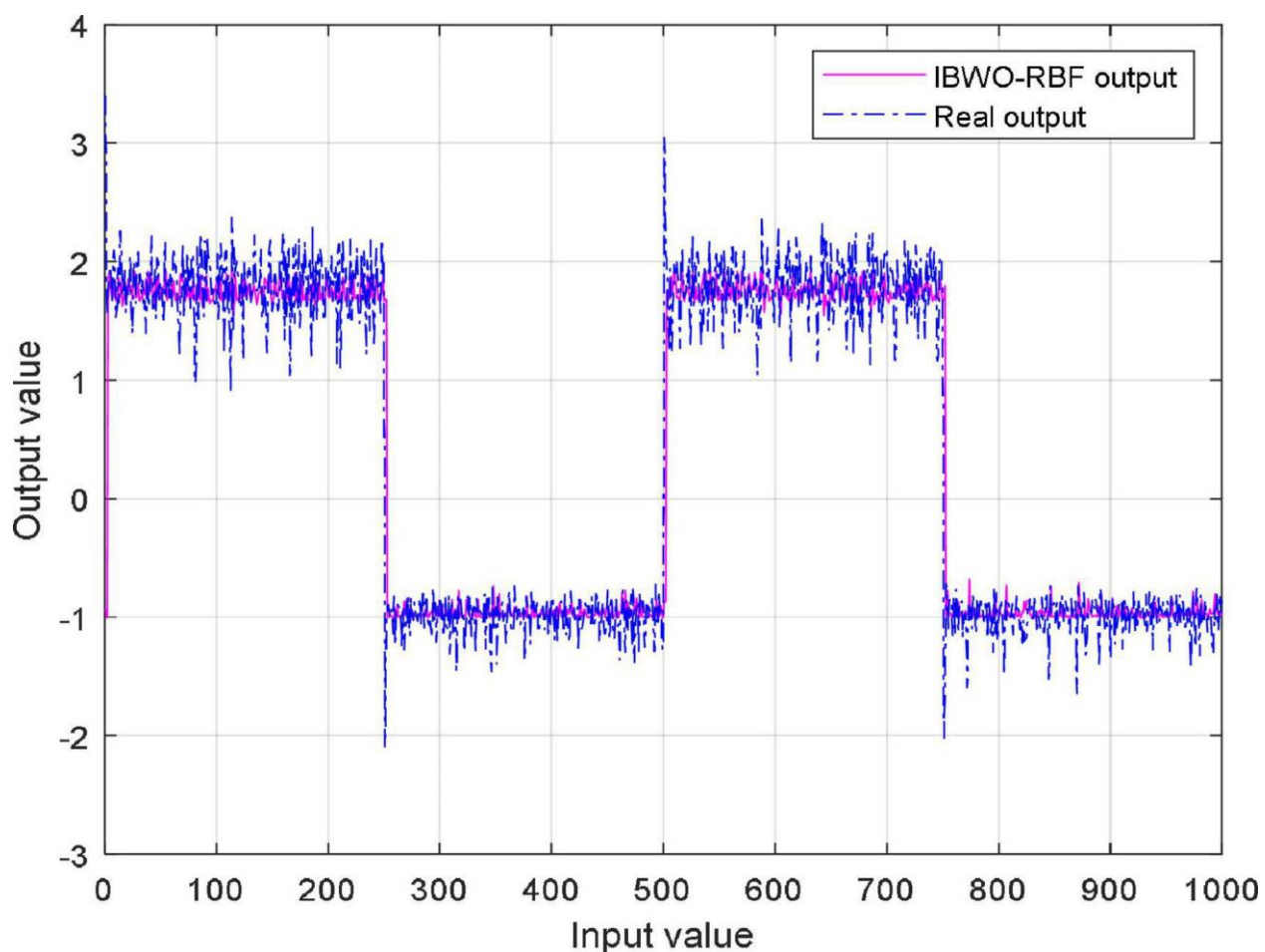


Рисунок 6 – Результаты идентификации алгоритма IBWO [7]

2.3 Использование RBF-сети в медико-экологических исследованиях

В статье [8] проводилась оценка эффективности применения искусственных нейронных сетей в медико-экологических исследованиях.

Поскольку все величины имели распределение, близкое к нормальному, результаты исследования представлены в виде среднего арифметического \pm стандартное отклонение ($M \pm SD$). Результаты тестирования были продемонстрированы в таблице 6.

Таблица 6 – Отношение стандартных отклонений ошибки прогноза и исходных данных изучаемых нейросетевых моделей. [8]

Класс болезней	РБС $M \pm SD$	МПС $M \pm SD$
Новообразования	$0,79 \pm 0,24$	$0,47 \pm 0,24$
Болезни крови	$0,87 \pm 0,19$	$0,65 \pm 0,25$
Болезни эндокринной системы	$0,86 \pm 0,22$	$0,62 \pm 0,21$
Болезни нервной системы	$0,88 \pm 0,23$	$0,74 \pm 0,24$
Бол. глаза и его придаточного апп.	$0,87 \pm 0,19$	$0,68 \pm 0,22$
Бол. уха и сосцевидного отростка	$0,87 \pm 0,20$	$0,65 \pm 0,26$
Болезни системы кровообращения	$0,90 \pm 0,22$	$0,79 \pm 0,33$
Болезни органов дыхания	$0,88 \pm 0,20$	$0,77 \pm 0,21$
Болезни органов пищеварения	$0,82 \pm 0,24$	$0,69 \pm 0,14$
Болезни кожи, подкожной клетчатки	$0,90 \pm 0,23$	$0,79 \pm 0,23$
Болезни костно-мышечной системы	$0,87 \pm 0,23$	$0,68 \pm 0,31$
Болезни мочеполовой системы	$0,90 \pm 0,21$	$0,74 \pm 0,14$
Отд. состояния в перинатальном периоде	$0,84 \pm 0,21$	$0,77 \pm 0,15$
Врожденные аномалии	$0,89 \pm 0,19$	$0,75 \pm 0,23$
Общий уровень	$0,82 \pm 0,20$	$0,67 \pm 0,24$

Из таблицы был сделан вывод, что наибольшим качеством прогноза обладает модель на основе радиально-базисных функций.

ЗАКЛЮЧЕНИЕ

Радиальные нейронные сети (RBF-сети) представляют собой мощный инструмент для решения широкого спектра задач, связанных с обработкой данных, распознаванием образов и классификацией. Проведенный в реферате анализ теоретических основ RBF-сетей, включая математическую базу и алгоритмы обучения демонстрирует их уникальные преимущества и универсальность.

Сравнительный анализ RBF-сетей и многослойных персептронов подчеркнул плюсы и минусы радиальных сетей.

Были рассмотрены исследования в области решения различных задач с использованием различных алгоритмов радиальных нейронных сетей. Полученные данные показали конкурентоспособность данных алгоритмов в сравнении с другими аналогами.

В заключение можно отметить, что радиальные нейронные сети остаются актуальным инструментом машинного обучения. Их дальнейшее развитие и адаптация к современным требованиям, включая интеграцию с передовыми алгоритмами оптимизации и обработкой больших данных, будут способствовать росту их востребованности в различных научных и прикладных областях.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Козин, Н. Е. Поэтапное обучение радиальных нейронных сетей / Н. Е. Козин, В. А. Фурсов. // Компьютерная оптика. – 2004. – № 26. – С. 138-141.
- 2 Глинский, И. В. Искусственные нейронные сети на основе радиально-базисных функций / И. В. Глинский // Компьютерные системы и сети : сборник статей 59-й научной конференции аспирантов, магистрантов и студентов / Белорусский государственный университет информатики и радиоэлектроники. – 2023. – С. 525–527.
- 3 Wurzberger, F. Learning in deep radial basis function networks / F. Wurzberger, F. Schwenker. // Entropy. – 2024. – № 26(5). – С. 368.
- 4 Грибанов, К.Г. Нейронные сети [Электронный ресурс] // Лаборатория физики климата и окружающей среды ИЕНиМ УрФУ [Электронный ресурс] : научный сайт. – URL: http://remotesensing.ru/main_r.html (дата обращения 20.12.24) Загл. с экрана. Яз. рус.
- 5 Хайкин, С. Нейронные сети. Полный курс : монография / С. Хайкин, [пер. с англ. Н.Н. Куссуль, А.Ю. Шелестова]. – 2-е изд., пер. с англ. – М.: Издательский дом “Вильямс”, 2006. – 1104 с.
- 6 Schwenker, F. Three learning phases for radial-basis-function networks / F. Schwenker, H.A. Kestler, G. Palm. // Neural Networks. – 2001. – № 14(4-5). – С. 439-458.
- 7 Lui, H. An RBF neural network based on improved black widow optimization algorithm for classification and regression problems / H. Lui, G. Zhou, Y. Zhou, H. Huang, X. Wei // Frontiers in neuroinformatics. – 2023. – № 16.
- 8 Петров, С. Б. Оценка эффективности применения искусственных нейронных сетей в медико-экологических исследованиях / С.Б. Петров, И.В. Шешунов. // Фундаментальные исследования. – 2013. – № 9-6.
- 9 Ganev, I. Universal approximation and model compression for radial neural networks [Электронный ресурс] // Arxiv [Электронный ресурс] : научный

сайт-архив. – URL: <https://arxiv.org/abs/2107.02550> (дата обращения 25.12.2024) Загл. с экрана. Яз. англ.