

ИНФОРМАТИКА,
ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА
И УПРАВЛЕНИЕ

INFORMATICS,
COMPUTER ENGINEERING
AND CONTROL

УДК 004.85

DOI: 10.17212/2782-2001-2021-2-67-82

Построение робастных нейронных сетей с различными функциями потерь^{*}

М.А. СИВАК^а, В.С. ТИМОФЕЕВ^б

630073, РФ, г. Новосибирск, пр. Карла Маркса, 20, Новосибирский государственный технический университет

^а pepelyaeva@ami.nstu.ru ^б v.timofeev@corp.nstu.ru

В статье рассмотрена задача построения устойчивых модификаций нейронных сетей с использованием различных робастных функций потерь. Применение таких нейронных сетей целесообразно при работе с зашумленными данными и может служить альтернативой предварительной очистке выборки или усложнению архитектуры сети. Для корректной работы алгоритма обучения нейронной сети (алгоритма обратного распространения ошибки) требуется, чтобы функция потерь была непрерывно или бесконечно дифференцируема. В соответствии с этим ограничением было выбрано пять робастных функций потерь – Эндрюса, Уэлша, Хьюбера, Рамсея и Faig. Использование этих функций в алгоритме обратного распространения ошибки вместо квадратичной функции потерь позволило получить абсолютно новый класс нейронных сетей. Для исследования свойств построенных сетей проводился ряд вычислительных экспериментов при различной доле засоряющих наблюдений в выборке и различном числе эпох обучения. На первом этапе производилась настройка полученных сетей, то есть выбор таких значений внутренних параметров функции потерь, при которых достигается наивысшая точность работы нейронной сети. Для определения интервалов, на которых рассматривались значения параметров, а также шага разбиения этих интервалов проводилось предварительное исследование. Полученные на первом этапе результаты позволили дать рекомендации по выбору наилучших значений параметров для каждой из рассмотренных функций. На втором этапе проводилось сравнение точности работы построенных робастных сетей как между собой, так и с классической нейронной сетью. Анализ результатов показал, что использование робастного подхода позволяет получить значительный выигрыш в точности работы и в скорости обучения нейронной сети.

Ключевые слова: искусственная нейронная сеть, алгоритм обратного распространения ошибки, выбросы, вычислительный эксперимент, робастный подход, функция потерь, машинное обучение, задача классификации

^{*} Статья получена 18 декабря 2020 г.

Исследование выполнено при финансовой поддержке РФФИ в рамках научного проекта № 20-37-90077.

ВВЕДЕНИЕ

Задача распознавания образов является одной из классических и наиболее распространенных задач машинного обучения, и зачастую для ее решения применяются искусственные нейронные сети (ИНС). Однако на практике исследователи неизбежно сталкиваются с тем, что имеющиеся данные содержат в себе некоторое количество нетипичных или засоряющих наблюдений (выбросов). Присутствие даже небольшого их числа может сильно сказаться на качестве работы нейронной сети с простой архитектурой [1–3].

Чтобы решить эту проблему, можно воспользоваться несколькими подходами. Самым очевидным вариантом является выполнение предобработки данных и исключение из них этих самых выбросов, однако такой подход может привести к потере важной информации. Кроме того, можно усложнить архитектуру ИНС, но в этом случае появятся дополнительные затраты на вычислительные ресурсы. Альтернативой этим двум методам может служить робастный подход [4, 5], использование которого позволит учесть нетипичные наблюдения и при необходимости снизить их негативное влияние в ходе обучения нейронной сети. В этом случае предполагается введение робастной функции потерь в алгоритме обучения нейронной сети вместо обычно используемой квадратичной функции. В рамках настоящей статьи рассматриваются несколько таких модификаций нейронной сети, полученных за счет использования различных робастных функций потерь.

1. ПОСТАНОВКА ЗАДАЧИ

Рассмотрим в общем виде задачу обучения нейронной сети. Для начала пусть имеется ИНС с одним скрытым слоем, ее архитектура представлена на рис. 1.

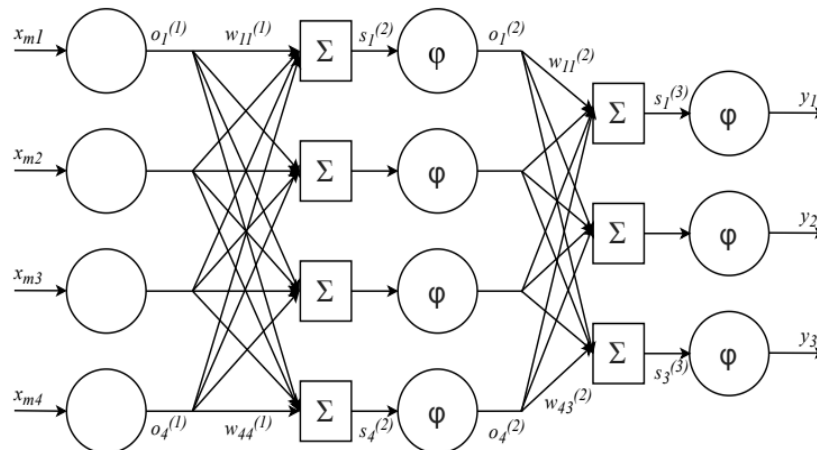


Рис. 1. Искусственная нейронная сеть с одним скрытым слоем

Fig. 1. An artificial neural network with one hidden layer

На этом рисунке использованы следующие обозначения: x_{mi} , $i=1, \dots, 4$ – значения, которые подаются на вход сети; y_1, y_2, y_3 – значения нейронов

выходного слоя сети; $w_{ij}^{(1)}$, $i, j = 1, \dots, 4$, $w_{jk}^{(2)}$, $k = 1, \dots, 3$ – веса на ребрах между нейронами соседних слоев (первого и второго, второго и третьего соответственно); $s_j^{(2)}$, $s_k^{(3)}$ – входные значения нейронов на втором и третьем слоях соответственно; $o_i^{(1)}$, $o_j^{(2)}$ – выходные значения нейронов на первом и втором слоях соответственно, φ – функция активации. В рамках настоящей работы в качестве функции активации $\varphi = \varphi(z)$ используется сигмоида [6]

$$\varphi(z) = \frac{1}{1 + e^z}.$$

Приведенные выше обозначения без труда можно обобщить на случай ИНС с более сложной архитектурой: x_{mi} , $i = 1, \dots, T$ – значения, которые подаются на вход сети; y_k , $k = 1, \dots, Y$ – значения на выходном слое нейронной сети; $w_{ij}^{(n-1)}$, $i = 1, \dots, l^{(n-1)}$, $j = 1, \dots, l^{(n)}$ – вес между j -м нейроном слоя n и i -м нейроном слоя $n - 1$ ($l^{(n)}$ – количество нейронов на слое n); $o_j^{(n)}$ – выходное значение j -го нейрона на слое n , определяющееся следующим образом:

$$o_j^{(n)} = \varphi\left(s_j^{(n)}\right), \quad (1)$$

где $s_j^{(n)}$ – входное значение j -го нейрона на слое n . Если нейрон находится на входном слое сети, то для него будет справедливо

$$s_i^{(1)} = o_i^{(1)} = x_{mi}.$$

В противном случае входное значение нейрона на слое n определяется через выходные значения нейронов на слое $n - 1$ как

$$s_j^{(n)} = \sum_{i=1}^{l^{(n-1)}} w_{ij}^{(n-1)} o_i^{(n-1)}, \quad n = 2, 3, \dots, N, \quad (2)$$

где N – количество слоев нейронной сети. Значения на выходном слое сети будут вычисляться в соответствии с выражением

$$y_k = \varphi\left(s_k^{(N)}\right). \quad (3)$$

Наиболее распространенным алгоритмом обучения ИНС является алгоритм обратного распространения ошибки, подробное описание которого можно найти в [6]. Обучение нейронной сети заключается в решении задачи оптимизации вида

$$E = \sum_{j=1}^l f(t_j, y_j) \rightarrow \min_{w_{ij}^{(1)}, \dots, w_{ij}^{(N-1)}}, \quad (4)$$

где E – суммарная функция потерь; t_j – требуемый ответ на j -м нейроне выходного слоя сети. Обычно при обучении нейронной сети используется квадратичная функция потерь

$$f(t_j, y_j) = \frac{1}{2}(y_j - t_j)^2. \quad (5)$$

Далее будем называть классической нейронную сеть, в алгоритме обучения которой используется данная функция.

Однако квадратичная функция потерь является неустойчивой к выбросам, вследствие чего в рамках настоящей работы ставится задача при использовании различных робастных функций потерь получить устойчивые варианты алгоритма обучения ИНС, а также провести сравнительный анализ работы построенных робастных сетей. Кроме того, для каждой из исследуемых функций потерь необходимо определить, при каких значениях их внутренних параметров точность работы сети будет наилучшей.

2. РОБАСТНЫЙ АЛГОРИТМ ОБУЧЕНИЯ

Для решения задачи оптимизации (4) требуется вычислить производную функции потерь по весам нейронной сети. На основе (1) и (3) можно записать следующее цепное правило для вычисления частной производной суммарной функции потерь [6]:

$$E'_{ji}^{(n)} = \frac{\partial E}{\partial w_{ij}^{(n-1)}} = \frac{\partial E}{\partial o_j^{(n)}} \frac{\partial o_j^{(n)}}{\partial s_j^{(n)}} \frac{\partial s_j^{(n)}}{\partial w_{ij}^{(n-1)}}.$$

Выполнив несложные преобразования с учетом (1)–(3) и (5), данную производную можно записать в виде

$$E'_{ji}^{(n)} = \delta_j^{(n)} o_i^{(n-1)},$$

где $\delta_j^{(n)}$ будет вычисляться следующим образом:

$$\delta_j^{(n)} = \frac{\partial E}{\partial o_j^{(n)}} \frac{\partial o_j^{(n)}}{\partial s_j^{(n)}} = \begin{cases} \frac{\partial f(y_j, t_j)}{\partial y_j} y_j(1 - y_j), & n = N, \\ \left(\sum_{k=1}^{l^{(n+1)}} w_{jk}^{(n)} \delta_k^{(n+1)} \right) \phi(s_j^{(n)}) (1 - \phi(s_j^{(n)})) & \text{иначе.} \end{cases} \quad (6)$$

Алгоритм обратного распространения ошибки накладывает ограничение на используемую функцию потерь: она должна быть непрерывно или бесконечно дифференцируема. С учетом этого ограничения было выбрано пять робастных функций потерь [7–11], анализ которых представлен в работе [12]. Эти функции и их частные производные по выходу j -го нейрона на выходном слое приводятся в табл. 1 (здесь и далее β – параметр робастной функции потерь, принимающий неотрицательные значения).

Таблица 1

Table 1

Робастные функции потерь и их производные

Robust loss functions and their derivatives

Функция потерь	Производная функции потерь
Хьюбера	
$\begin{cases} \frac{1}{2}(y_j - t_j)^2, & y_j - t_j \leq \beta \\ \beta y_j - t_j - \frac{1}{2}\beta^2, & y_j - t_j > \beta \end{cases}$	$\begin{cases} y_j - t_j, & y_j - t_j \leq \beta \\ -\beta, & y_j - t_j < -\beta \\ \beta, & y_j - t_j > \beta \end{cases}$
Эндрюса	
$\begin{cases} \beta(1 - \cos \frac{(y_j - t_j)}{\beta}), & y_j - t_j < \pi\beta \\ 2\beta, & y_j - t_j \geq \pi\beta \end{cases}$	$\begin{cases} \sin \frac{(y_j - t_j)}{\beta}, & y_j - t_j < \pi\beta \\ 0, & y_j - t_j \geq \pi\beta \end{cases}$
Рамсея	
$\frac{1 - (1 + \beta y_j - t_j)\exp(-\beta y_j - t_j)}{\beta^2}$	$(y_j - t_j) \exp\{-\beta y_j - t_j \}$
Fair	
$\beta^2 \left(\frac{ y_j - t_j }{\beta} - \ln \left(1 + \frac{ y_j - t_j }{\beta} \right) \right)$	$\frac{y_j - t_j}{(1 + y_j - t_j /\beta)}$
Уэлша	
$1 - \exp \left(-\frac{1}{2} \left(\frac{y_j - t_j}{\beta} \right)^2 \right)$	$\frac{1}{\beta^2} (y_j - t_j) \exp \left(-\frac{1}{2} \left(\frac{y_j - t_j}{\beta} \right)^2 \right)$

Заменим (5) на одну из робастных функций потерь из табл. 1, обозначив ее как $f_R(t_j, y_j)$. Тогда (6) примет вид

$$\delta_j^{(n)} = \frac{\partial E}{\partial o_j^{(n)}} \frac{\partial o_j^{(n)}}{\partial s_j^{(n)}} = \begin{cases} \frac{\partial f_R(y_j, t_j)}{\partial y_j} y_j (1 - y_j), & n = N, \\ \left(\sum_{k=1}^{l^{(n+1)}} w_{jk}^{(n)} \delta_k^{(n+1)} \right) \varphi(s_j^{(n)}) (1 - \varphi(s_j^{(n)})) & \text{иначе.} \end{cases}$$

Таким образом, проводя обучение ИНС с использованием этих функций потерь, получим класс совершенно новых нейронных сетей. Далее приводятся результаты исследований их свойств.

3. РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЙ

Сравнение точности работы полученных ИНС проводилось на наборе данных «Ирисы Фишера» [13], который включает в себя 150 объектов X_m , $m = 1, \dots, 150$, в равных долях принадлежащих трем различным классам y_k ,

$k = 1, \dots, 3$, и описываемых четырьмя признаками x_{mi} , $i = 1, \dots, 4$. Выбранный набор данных является сравнительно небольшим – за счет этого появляется возможность отслеживать закономерности в данных и анализировать результат работы сети.

При проведении исследований описанный набор данных делился на обучающую и тестовую выборки. Обучающая выборка $L = \{X_1, \dots, X_{|L|}\}$ – это набор объектов, используемый для обучения нейронной сети. В данном случае в нее вошло 80 % данных исходного набора ($|L| = 120$ объектов – объем обучающей выборки). Оставшиеся 20 % объектов вошли в тестовую выборку $D = \{X_{|L|+1}, \dots, X_{|X|}\}$, которая использовалась для оценки эффективности работы нейронной сети ($|D| = 30$ объектов – объем тестовой выборки).

В данном случае все классы являются сбалансированными, поэтому для оценки точности работы построенных нейронных сетей использовалась только метрика *accuracy* [14] – отношение числа правильно классифицированных объектов тестовой выборки к объему тестовой выборки:

$$\alpha = \frac{D_{corr}}{|D|} 100 \%,$$

где D_{corr} – количество объектов, отнесенных к правильному классу.

На рис. 2 для каждой пары признаков приводятся диаграммы рассеивания, позволяющие наглядно увидеть влияние признаков на взаимное расположение классов.

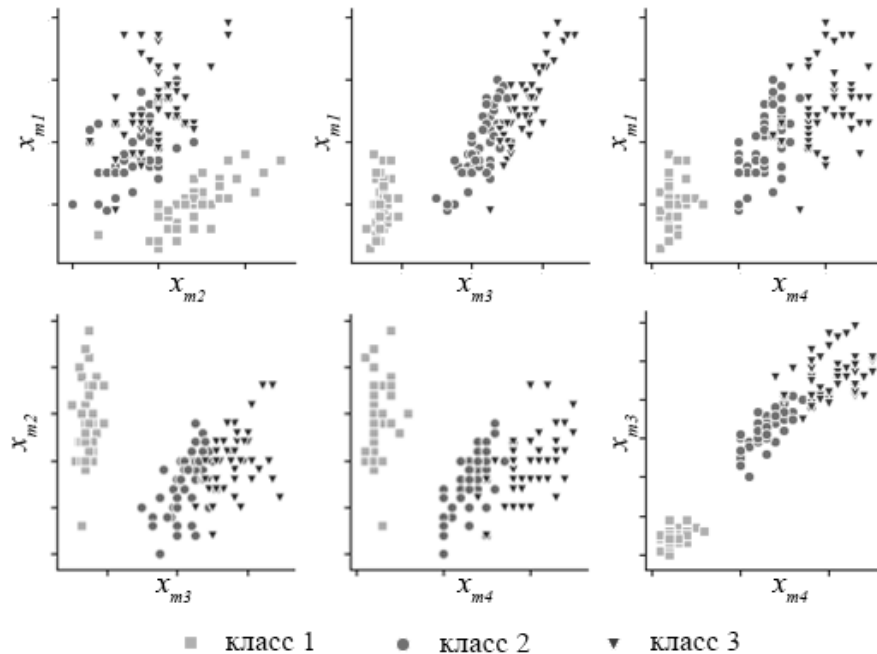


Рис. 2. Попарные графики признаков объектов набора «Ирисы Фишера»

Fig. 2. Pair plots of the Fisher Iris observations attributes

Из рис. 2 видно, что наиболее важными с точки зрения различия объектов являются третий и четвертый признаки, поэтому зашумление обучающей выборки выполнялось по ним:

$$\tilde{x}_{mi} = x_{mi} + \varepsilon_{mi}, \quad i = 3, 4,$$

где ε_{mi} – случайные, независимые, одинаково распределенные ошибки. Эти ошибки имеют следующую функцию распределения:

$$F_i(x) = (1 - \lambda)F_1(x, 0, \sigma_{i1}^2) + \lambda F_2(x, 0, \sigma_{i2}^2), \quad i = 3, 4,$$

где $F_j(x, 0, \sigma_{ij}^2)$, $j = 1, 2$, – функция нормального распределения с нулевым математическим ожиданием и дисперсией σ_{ij}^2 , $\lambda \in [0, 1]$ – параметр смеси, играющий роль доли засоряющих наблюдений.

В рамках настоящей работы полагалось, что $\sigma_{i1}^2 < \sigma_{i2}^2$, причем задавались не сами значения дисперсий σ_{i1}^2 и σ_{i2}^2 , а соответствующие им значения уровня шума. Уровень шума [15] определялся как

$$\rho_{ij} = \frac{\sigma_{ij}^2}{c} 100 \%,$$

где c^2 – дисперсия незашумленной выборки.

При проведении исследований для третьего признака дисперсия σ_{31}^2 соответствовала уровню шума $\rho_{31} = 30 \%$, а дисперсия σ_{32}^2 – уровню шума $\rho_{32} = 120 \%$; для четвертого признака дисперсия σ_{41}^2 соответствовала уровню шума $\rho_{41} = 40 \%$, а дисперсия σ_{42}^2 – уровню шума $\rho_{42} = 150 \%$. Значение λ доли засоряющих наблюдений в обучающей выборке варьировалось в пределах от 0,05 до 0,40 с шагом в 0,05.

На первом этапе исследований необходимо было выполнить настройку полученных робастных нейронных сетей, то есть определить наилучшие значения параметра робастных функций потерь β . Иными словами, это такие значения, при которых точность работы сети была бы наилучшей; значения параметра варьировались на различных интервалах $(\beta_{\min}, \beta_{\max}]$.

Для определения этих интервалов было проведено предварительное исследование. Для всех функций из табл. 1 были зафиксированы левая граница интервала $\beta_{\min} = 0,00$, правая граница $\beta_{\max} = 15,00$ и шаг разбиения 0,10. Значение доли засоряющих наблюдений было зафиксировано в точке $\lambda = 0,25$. Для каждого значения β на этом интервале проводилось по 10 вычислительных экспериментов, в ходе которых фиксировалось значение метрики α на 500 эпох обучения. Полученные результаты затем усреднялись.

Проведенное исследование позволило определить правую границу интервала – такое значение β , при котором точность работы сети перестает значительно изменяться. Кроме того, для функций потерь Хьюбера и Fair был скорректирован шаг разбиения интервала. В первом случае шаг разбиения 0,10 оказался большим для выбранного интервала, вследствие этого не всегда

удавалось найти наилучшее значение параметра. Во втором случае данный шаг разбиения, наоборот, оказался слишком маленьким – такое разбиение привело бы к значительному увеличению вычислительных затрат при поиске наилучших значений β . Полученные в результате предварительного исследования границы и шаг разбиения интервалов приводятся в табл. 2.

Таблица 2

Table 2

Интервалы, на которых рассматривались значения параметра робастных функций потерь

Ranges of the robust loss function parameter values

Функция потерь	β_{\min}	β_{\max}	Шаг разбиения
Эндрюса	0,00	5,00	0,10
Уэлша	0,00	8,00	0,10
Хьюбера	0,00	1,00	0,05
Рамсея	0,00	5,00	0,10
Fair	0,00	10,00	0,50

Далее исследования проводились следующим образом. Для всех обозначенных выше значений доли засоряющих наблюдений при различном числе эпох обучения нейронной сети (50, 100 и далее до 1000 с шагом в 100 эпох) фиксировалось значение метрики α , полученное для каждого из значений параметра β на интервалах из табл. 2. Соотношение количества эпох обучения и значения метрики α позволяет судить о скорости обучения ИНС (очевидно, что чем меньше количество эпох и чем больше при этом значение α , тем быстрее обучается нейронная сеть).

Для всех робастных сетей выполнялось по 100 вычислительных экспериментов, результаты которых затем усреднялись. После этого для каждой исследуемой ИНС были выбраны наилучшие значения параметра функции потерь β . В табл. 3 приводятся эти значения при различной доле выбросов на 100, 300, 500 и 1000 эпох обучения.

Таблица 3

Table 3

Значения параметра робастной функции потерь, при которых достигается наилучшая точность работы сетей

Values of the robust loss functions parameters that maximize the accuracy of neural networks

λ	Число эпох	Функция потерь				
		Эндрюса	Уэлша	Хьюбера	Рамсея	Fair
0,05	100	1,5	1,1	0,55	0,6	1,5
	300	5,0	2,3	0,5	0,8	1,5
	500	5,0	3,1	0,45	1,4	0,5
	1000	5,0	3,5	0,3	1,7	0,1
0,10	100	1,6	1,2	0,55	0,6	1,0
	300	5,0	2,6	0,25	1,6	0,5

Окончание табл. 3

End of Tab. 3

λ	Число эпох	Функция потерь				
		Эндрюса	Уэлша	Хьюбера	Рамсея	Fair
	500	5,0	3,2	0,25	1,7	0,5
	1000	5,0	3,9	0,2	1,8	0,1
0,15	100	1,7	1,2	0,45	0,8	1,0
	300	4,9	2,4	0,3	1,4	0,5
	500	5,0	3,2	0,25	1,7	0,5
	1000	5,0	3,7	0,25	1,7	0,1
0,20	100	4,8	2,1	0,4	1,1	0,5
	300	5,0	2,5	0,25	1,6	0,5
	500	5,0	3,2	0,2	1,7	0,5
	1000	5,0	4,1	0,25	1,7	0,1
0,25	100	2,3	1,5	0,4	0,9	0,5
	300	5,0	2,5	0,25	1,6	0,5
	500	5,0	2,9	0,25	1,7	0,1
	1000	5,0	3,7	0,3	1,7	0,1
0,30	100	2,4	1,5	0,45	0,9	0,5
	300	4,7	2,2	0,25	1,6	0,5
	500	5,0	2,7	0,25	1,6	0,5
	1000	5,0	4,4	0,35	1,4	0,5
0,35	100	2,9	1,6	0,4	1,0	0,5
	300	5,0	2,4	0,25	1,6	0,5
	500	5,0	3,1	0,25	1,7	0,5
	1000	5,0	3,7	0,25	1,5	0,5
0,40	100	2,5	0,5	0,4	1,0	0,5
	300	5,0	2,4	0,25	1,6	0,5
	500	5,0	2,8	0,25	1,6	0,5
	1000	5,0	4,6	0,35	1,5	0,1

Приведенные результаты показывают, что практически у всех функций потерь, за исключением функции Хьюбера, явная зависимость между значением параметра β и долей засоряющих наблюдений фактически не прослеживается. Для функции потерь Хьюбера при сравнительно небольшой доле выбросов ($\lambda \leq 0,15$) характерно убывание значения β с ростом числа эпох обучения. При большей доле выбросов значение параметра с ростом числа эпох начинает колебаться либо не изменяется ($\lambda = 0,35$).

Для остальных рассмотренных функций наблюдается зависимость значения параметра от числа эпох обучения. Так, в случае функций потерь Эндрюса и Уэлша с ростом числа эпох значение параметра, при котором достигается наилучшая точность работы сети, возрастает. Для нейронной сети с функцией потерь Рамсея при доле выбросов $\lambda \leq 0,25$ наилучшее значение β возрастает с ростом числа эпох. При большей доле выбросов и числе эпох обучения до 500 оно так же возрастает, а при числе эпох от 501 до 1000, наоборот, начинает убывать. В случае функции потерь Fair наилучшее значе-

ние параметра с ростом числа эпох обучения чаще всего убывает (либо не изменяется при $\lambda = 0,30$ и $\lambda = 0,35$).

Зависимость наилучшего значения параметра β от числа эпох обучения особенно ярко выражена для функции потерь Уэлша, что продемонстрировано на рис. 3, где показана зависимость точности работы сети от значения параметра β при различном числе эпох для доли засоряющих наблюдений 0,25.

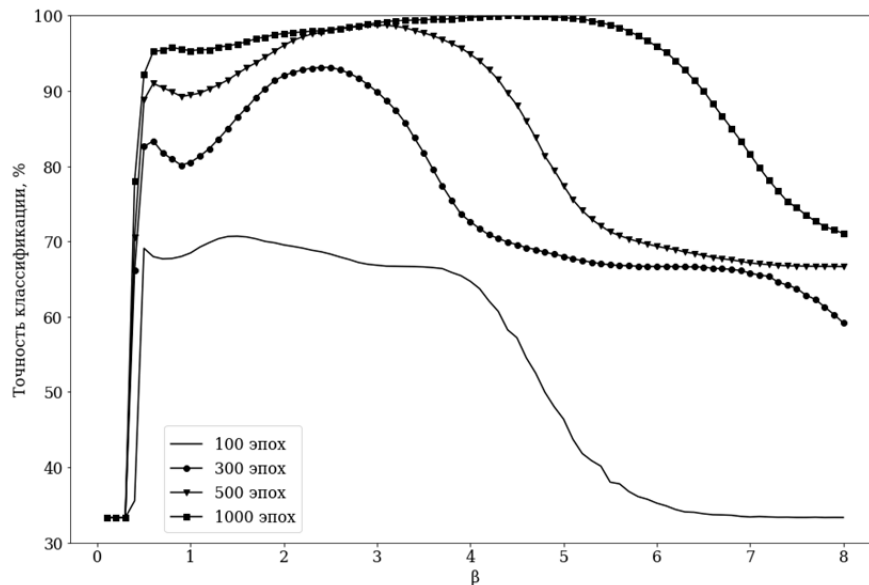


Рис. 3. Зависимость наилучшего значения параметра функции потерь Уэлша от числа эпох обучения

Fig. 3. Dependency of the optimal value of the Welsch loss function parameter on the learning epoch number

На основе полученных результатов можно дать следующие рекомендации относительно выбора значений параметра β для исследуемых нейронных сетей:

- функция потерь Эндрюса: до 100 эпох обучения — на отрезке $[1,5; 2,9]$, более 100 эпох — на отрезке $[4,7; 5,0]$;
- функция потерь Уэлша: до 300 эпох обучения — на отрезке $[0,5; 2,1]$, от 301 до 500 эпох обучения — на отрезке $[2,3; 3,2]$, более 500 эпох обучения — на отрезке $[3,5; 4,6]$;
- функция потерь Хьюбера: при малой доле выбросов ($\lambda \leq 0,05$) или при малом числе эпох (до 100) — на отрезке $[0,4; 0,55]$, при числе эпох больше 100 или при большей доле выбросов ($\lambda \geq 0,10$) — на отрезке $[0,2; 0,35]$;
- функция потерь Рамсея: до 100 эпох обучения — на отрезке $[0,6; 1,0]$, больше 100 эпох — на отрезке $[0,8; 1,7]$;
- функция потерь Fair: до 300 эпох обучения — на отрезке $[0,5; 1,5]$, более 300 эпох обучения — на отрезке $[0,1; 0,5]$.

На втором этапе исследований проводилось сравнение точности работы полученных робастных ИНС между собой, а также с классической сетью. Для этого рассматривались значения метрики α , полученные при наилучших

значениях параметра β . В табл. 4 приводятся результаты для всех обозначенных ранее значений доли засоряющих наблюдений на отметках в 100, 300 и 500 эпох обучения. При большем числе эпох точность работы всех сетей резко возрастала и в отдельных случаях достигала значений 98...99 %, что однозначно говорит о переобучении. При сравнении точности работы построенных сетей рассматривать такие случаи нецелесообразно.

Таблица 4

Table 4

**Точность работы нейронных сетей с различными функциями потерь
при разных значениях доли выбросов**

Accuracy of neural networks with different loss functions for various outlier fraction values

λ	Число эпох	Функция потерь					
		Эндрюса	Уэлша	Хьюбера	Рамсея	Fair	Квадратичная
0,05	100	76,6	77,0	76,7	77,1	76,9	75,0
	300	91,2	91,3	90,7	90,4	90,4	89,2
	500	96,6	96,8	94,3	95,0	95,1	91,4
0,10	100	72,3	72,3	71,9	72,2	72,2	71,6
	300	89,4	89,8	83,2	83,9	83,2	77,8
	500	92,7	95,6	91,3	91,6	89,8	83,5
0,15	100	71,0	71,3	71,5	71,5	71,5	69,8
	300	86,2	86,6	84,8	84,6	84,4	77,8
	500	91,0	94,3	90,6	90,2	88,8	82,4
0,20	100	68,2	68,0	67,7	68,0	68,0	67,2
	300	79,5	79,9	78,6	78,7	77,9	72,6
	500	88,7	91,2	89,1	88,9	86,0	78,9
0,25	100	70,7	70,7	69,3	70,0	70,1	67,3
	300	92,7	93,1	87,3	88,3	87,1	76,6
	500	97,1	98,7	94,0	94,7	94,0	85,2
0,30	100	69,0	69,1	67,3	68,1	68,4	66,3
	300	87,4	87,6	85,1	86,3	85,5	76,4
	500	93,9	95,5	92,2	92,7	91,7	84,8
0,35	100	69,6	69,6	67,7	68,7	68,9	65,6
	300	88,5	89,5	84,1	84,2	82,4	74,7
	500	92,5	96,0	90,8	91,3	90,0	81,8
0,40	100	69,3	69,4	69,0	69,2	69,1	67,0
	300	86,3	87,0	84,9	85,7	82,8	74,3
	500	93,7	95,4	90,7	91,8	89,9	80,3

Из табл. 4 видно, что при сравнительно малом числе эпох (100) точность работы, а следовательно, и скорость обучения робастных нейронных сетей сопоставимы с показателями классической нейронной сети. Кроме того, можно заметить, что при малой доле засоряющих наблюдений ($\lambda = 0,05$) даже с увеличением числа эпох до 300 робастные нейронные сети работают незна-

чительно точнее классической (на 1,6 % в среднем), а если увеличить число эпох до 500, то робастные сети начинают работать точнее в среднем на 4,2 %.

Однако с возрастанием доли засоряющих наблюдений до 0,10 уже на 300 эпохах нейронные сети, при построении которых используются робастные функции потерь, обучаются быстрее и показывают точность в среднем на 8,1 % выше, чем классическая нейронная сеть. При работе с сильно зашумленными выборками (доля выбросов $\lambda = 0,35$ и $\lambda = 0,40$) робастные сети работают точнее в среднем на 11 % и при 300 эпохах обучения, и при 500.

Для большей наглядности на рис. 4 приводятся графики точности работы для всех исследуемых нейронных сетей на 500 эпох обучения при различной доле засоряющих наблюдений. Данные графики позволяют сделать вывод о том, что все робастные сети дают значительный выигрыш в точности по сравнению с классической.

Нетрудно заметить, что нейронная сеть, в алгоритме обучения которой использовалась функция потерь Уэлша, работает точнее остальных НС при всех значениях доли засоряющих наблюдений. По сравнению с классической ИНС при $\lambda = 0,25$ выигрыш в точности на 500 эпох обучения составляет 13,5 %, а при сильном зашумлении выборки (при $\lambda = 0,35$ и $\lambda = 0,40$) выигрыш составляет 14,2 % и 15,1 % соответственно.

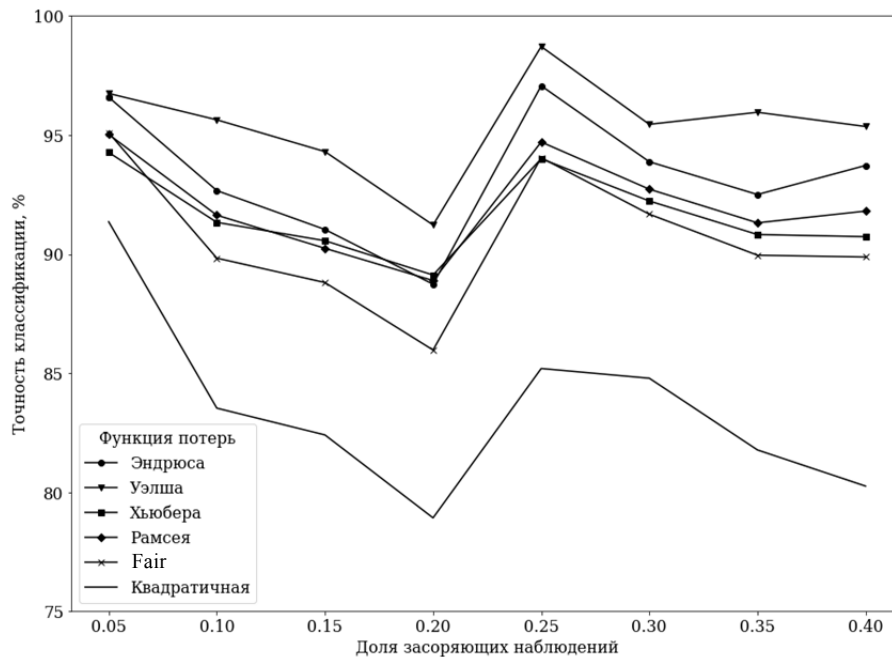


Рис. 4. Точность работы исследуемых НС при различной доле выбросов на 500 эпох

Fig. 4. Accuracy of the investigated neural networks for various outlier fraction values and for 500 epochs

Кроме того, видно, что нейронная сеть, построенная с использованием функции потерь Fair, при значениях λ от 0,10 до 0,25 работает хуже других робастных сетей, но с ростом доли выбросов точность ее работы становится сопоставима с сетями, в алгоритме обучения которых использовались функ-

ции Рамсея и Хьюбера. Нейронная сеть с функцией потерь Эндрюса, наоборот, в большинстве случаев работает несколько лучше нейронных сетей, построенных на основе функций Рамсея, Хьюбера и Fair, и лишь при значении $\lambda = 0,20$ точность ее работы становится сопоставима с точностью работы перечисленных выше трех робастных сетей.

Помимо всего прочего, полученные результаты позволяют отчетливо увидеть ухудшение точности работы всех ИНС – и классической, и робастных – при доле засоряющих наблюдений $\lambda = 0,20$, а также резкое улучшение точности при доле выбросов $\lambda = 0,25$ и последующее ее ухудшение с ростом значения доли выбросов. Этот парадокс можно объяснить конфигурацией выборок, а именно тем, насколько различные классы объектов удалены друг от друга, если за расстояние между классами принять расстояние между центрами масс точек, соответствующих объектам каждого класса. Поскольку зашумление производилось только по 3-му и 4-му признакам объектов, то точки, соответствующие объектам, рассматривались в двумерном евклидовом пространстве: первая координата точки соответствовала значению третьего признака, вторая – значению четвертого. Значение этого расстояния было получено для каждой из ста выборок при значениях доли выбросов $\lambda = 0,20$ и $\lambda = 0,25$. Вычисленные значения расстояния затем усреднялись. Полученные результаты представлены в табл. 5.

Таблица 5

Table 5

Расстояния между классами объектов при различной доле засоряющих наблюдений

Distances between object classes for various outlier fraction values

λ	Расстояние между классами		
	1-й и 2-й классы	2-й и 3-й классы	1-й и 3-й классы
0,20	3,098	1,399	4,484
0,25	2,946	1,686	4,624

Нетрудно заметить, что с ростом доли выбросов расстояние между первым и вторым классами уменьшилось на 4,9 %, а между первым и третьим классами увеличилось на 3 %. В то же время расстояние между вторым и третьим классами увеличилось на 20,5 %. Исследования показали, что при $\lambda = 0,20$ ИНС гораздо чаще «путают» объекты этих классов между собой, чем при $\lambda = 0,25$, что и приводит к такому резкому перепаду в точности работы сетей. Фактически ситуация, когда значение доли засоряющих наблюдений равно 0,20, в данном случае эквивалентна большей степени зашумления выборки.

ЗАКЛЮЧЕНИЕ

При выполнении исследований были получены пять робастных нейронных сетей с использованием различных функций потерь (Эндрюса, Уэлша, Рамсея, Хьюбера и Fair). Были исследованы свойства построенных ИНС.

Для этого был выполнен ряд вычислительных экспериментов, в ходе которых рассматривались различные значения доли выбросов, а значение точности работы нейронных сетей фиксировалось при различном числе эпох обучения.

Проведенные исследования позволили сформировать рекомендации по выбору наилучшего значения параметра β для каждой из рассмотренных функций потерь. Кроме того, полученные результаты показали, что при достаточно малой доле выбросов точность работы робастных нейронных сетей сопоставима с классической ИНС. Однако с ростом доли засоряющих наблюдений в выборке применение робастного подхода позволяет добиться значительного выигрыша как в скорости обучения нейронной сети, так и в точности ее работы.

СПИСОК ЛИТЕРАТУРЫ

1. Ланкин Ю.П., Басканова Т.Ф., Лобова Т.И. Нейросетевой анализ сложноорганизованных экологических данных // Современные проблемы науки и образования. – 2012. – № 4. – URL: <https://www.science-education.ru/ru/article/view?id=6754> (дата обращения: 01.12.2020).
2. Манжула В.Г., Федяшов Д.С. Нейронные сети Кохонена и нечеткие нейронные сети в интеллектуальном анализе данных // Фундаментальные исследования. – 2011. – № 4. – С. 108–115. – URL: <https://www.fundamental-research.ru/ru/article/view?id=21239> (дата обращения: 05.12.2020).
3. Глубокие нейросети (Ч. 1). Подготовка данных. – URL: <https://www.mql5.com/ru/articles/3486> (дата обращения: 26.11.2020).
4. Fan J., Gijbels I. Local polynomial modelling and its applications. – London: Chapman & Hall, 1996. – 360 p. – DOI: 10.1201/9780203748725.
5. Huber J.P. Robust statistics. – 2nd ed. – Hoboken, NJ: Wiley, 2009. – 370 p. – DOI: 10.1002/9780470434697.
6. Bishop C. Neural networks for pattern recognition. – New York: Oxford University Press, 1995. – 502 p.
7. Айвазян С.А., Енюков И.С., Мешалкин Л.Д. Прикладная статистика: исследование зависимостей. – М.: Финансы и статистика, 1985. – 448 с.
8. Вучков И., Бояджиева Л., Солаков Е. Прикладной линейный регрессионный анализ. – М.: Финансы и статистика, 1987. – 240 с.
9. Демиденко Е.З. Линейная и нелинейная регрессии. – М.: Финансы и статистика, 1981. – 304 с.
10. Black M.J., Rangarajan A. On the unification of line processes, outlier rejection, and robust statistics with applications in early vision // International Journal of Computer Vision. – 1996. – Vol. 19. – P. 57–91. – DOI: 10.1007/BF00131148.
11. Fair loss: margin-aware reinforcement learning for deep face recognition / B. Liu, W. Deng, Y. Zhong, M. Wang, J. Hu, X. Tao, Y. Huang // 2019 IEEE/CVF International Conference on Computer Vision (ICCV). – Seoul, Korea (South), 2019. – P. 10051–10060. – DOI: 10.1109/ICCV.2019.01015.
12. Сивак М.А. Исследование применимости робастных функций потерь в нейронных сетях // Сборник научных трудов НГТУ. – 2020. – № 4 (99). – С. 50–58. – DOI: 10.17212/2307-6879-2020-4-50-58.
13. UCI Machine Learning Repository. – URL: <http://www.ics.uci.edu/ml/MLRepository.html> (accessed: 03.05.2021).
14. Classification: Accuracy. – URL: <https://developers.google.com/machine-learning/crash-course/classification/accuracy> (accessed: 03.05.2021).
15. Ивахненко А.Г., Степашико В.С. Помехоустойчивость моделирования. – Киев: Наукова думка, 1985. – 216 с.

Сивак Мария Алексеевна, аспирант кафедры теоретической и прикладной информатики Новосибирского государственного технического университета. Основное направление научных исследований – прикладная математическая статистика, машинное обучение, обработка текстов на естественном языке. E-mail: pepelyaeva@ami.nstu.ru

Тимофеев Владимир Семенович, доктор технических наук, профессор кафедры теоретической и прикладной информатики Новосибирского государственного технического университета. Основное направление научных исследований – разработка и исследование устойчивых методов и алгоритмов анализа многофакторных объектов, в том числе с использованием непараметрической статистики. E-mail: v.timofeev@corp.nstu.ru

Sivak Maria A., a postgraduate student at the department of theoretical and applied computer science, Novosibirsk State Technical University. The focus of her research is applied mathematical statistics, machine learning, and natural language processing. E-mail: pepelyaeva@ami.nstu.ru

Timofeev Vladimir S., D.Sc. (Eng.), professor of the department of theoretical and applied computer science, Novosibirsk State Technical University. The focus of his research is developing and investigating robust methods and algorithms for a multifactor objects analysis using distribution-free statistics techniques. E-mail: v.timofeev@corp.nstu.ru

DOI: 10.17212/2782-2001-2021-2-67-82

Building robust neural networks using different loss functions*

M.A. SIVAK¹, V.S. TIMOFEEV²

Novosibirsk State Technical University, 20 K. Marx Prospekt, Novosibirsk, 630073, Russian Federation

^a pepelyaeva@ami.nstu.ru ^b v.timofeev@corp.nstu.ru

Abstract

The paper considers the problem of building robust neural networks using different robust loss functions. Applying such neural networks is reasonable when working with noisy data, and it can serve as an alternative to data preprocessing and to making neural network architecture more complex. In order to work adequately, the error back-propagation algorithm requires a loss function to be continuously or two-times differentiable. According to this requirement, two five robust loss functions were chosen (Andrews, Welsch, Huber, Ramsey and Fair). Using the above-mentioned functions in the error back-propagation algorithm instead of the quadratic one allows obtaining an entirely new class of neural networks. For investigating the properties of the built networks a number of computational experiments were carried out. Different values of outliers' fraction and various numbers of epochs were considered. The first step included adjusting the obtained neural networks, which lead to choosing such values of internal loss function parameters that resulted in achieving the highest accuracy of a neural network. To determine the ranges of parameter values, a preliminary study was pursued. The results of the first stage allowed giving recommendations on choosing the best parameter values for each of the loss functions under study. The second stage dealt with comparing the investigated robust networks with each other and with the classical one. The analysis of the results shows that using the robust technique leads to a significant increase in neural network accuracy and in a learning rate.

Keywords: artificial neural network, error back-propagation algorithm, outliers, computational experiment, robust technique, loss function, machine learning, classification problem

* Received 18 December 2020.

REFERENCES

1. Lankin Yu.P., Baskanova T.F., Lobova T.I. Neurosetevoi analiz slozhnoorganizovannykh ekologicheskikh dannykh [Neural network analysis of complicated ecological data]. *Sovremennye problemy nauki i obrazovaniya* = *Modern problems of science and education*, 2012, no. 4. Available at: <https://www.scienceeducation.ru/ru/article/view?id=6754> (accessed 01.12.2020).
2. Manzhula V.G., Fedyashov D.S. Neironnye seti Kokhonena i nechetkie neironnye seti v intellektual'nom analize dannykh [Kohonen neural networks and fuzzy neural networks in data mining]. *Fundamental'nye issledovaniya* = *Fundamental research*, 2011, no. 4, pp. 108–115. Available at: <https://www.fundamentalresearch.ru/ru/article/view?id=21239> (accessed 05.12.2020).
3. *Glubokie neuroseti* (Ch. 1). *Podgotovka dannykh* [Deep neural networks (Pt. 1). Data preprocessing]. Available at: <https://www.mql5.com/ru/articles/3486> (accessed 26.11.2020).
4. Fan J., Gijbels I. *Local polynomial modelling and its applications*. London, Chapman & Hall, 1996. 360 p. DOI: 10.1201/9780203748725.
5. Huber J.P. *Robust statistics*. 2nd ed. Hoboken, NJ, Wiley, 2009. 370 p. DOI: 10.1002/9780470434697.
6. Bishop C. *Neural networks for pattern recognition*. New York, Oxford University Press, 1995. 502 p.
7. Aivazyan S.A., Enyukov I.S., Meshalkin L.D. *Prikladnaya statistika: issledovanie zavisimostei* [Applied statistics: dependency investigation]. Moscow, Finansy i statistika Publ., 1985. 448 p.
8. Vuchkov I., Boyadzhieva L., Solakov E. *Prikladnoi lineinyi regressingionnyi analiz* [Applied linear regression analysis]. Moscow, Finansy i statistika Publ., 1987. 240 p.
9. Demidenko E.Z. *Lineinaya i nelineinaya regressii* [Linear and non-linear regressions]. Moscow, Finansy i statistika Publ., 1981. 304 p.
10. Black M.J., Rangarajan A. On the unification of line processes, outlier rejection, and robust statistics with applications in early vision. *International Journal of Computer Vision*, 1996, vol. 19, pp. 57–91. DOI: 10.1007/BF00131148.
11. Liu B., Deng W., Zhong Y., Wang M., Hu J., Tao X., Huang Y. Fair loss: margin-aware reinforcement learning for deep face recognition. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea (South), 2019, pp. 10051–10060. DOI: 10.1109/ICCV.2019.01015.
12. Sivak M.A. Issledovanie primenimosti roblastnykh funktsii poter' v neironnykh setyakh [The research on using robust loss functions for neural networks]. *Sbornik nauchnykh trudov Novosibirskogo gosudarstvennogo tekhnicheskogo universiteta* = *Transaction of scientific papers of the Novosibirsk state technical university*, 2020, no. 4 (99), pp. 50–58. DOI: 10.17212/2307-6879-2020-4-50-58.
13. *UCI Machine Learning Repository*. Available at: <http://www.ics.uci.edu/ml/MLRepository.html> (accessed 03.05.2021).
14. *Classification: Accuracy*. Available at: <https://developers.google.com/machine-learning/crash-course/classification/accuracy> (accessed 03.05.2021).
15. Ivakhnenko A.G., Stepashko V.S. *Pomekhoustoichivost' modelirovaniya* [Noise-resistant modeling]. Kiev, Naukova dumka Publ., 1985. 216 p.

Для цитирования:

Сивак М.А., Тимофеев В.С. Построение робастных нейронных сетей с различными функциями потерь // Системы анализа и обработки данных. — 2021. — № 2 (82). — С. 67–82. — DOI: 10.17212/2782-2001-2021-2-67-82.

For citation:

Sivak M.A., Timofeev V.S. Postroenie roblastnykh neironnykh setei s razlichnymi funktsiyami poter' [Building robust neural networks using different loss functions]. *Sistemy analiza i obrabotki dannykh* = *Analysis and Data Processing Systems*, 2021, no. 2 (82), pp. 67–82. DOI: 10.17212/2782-2001-2021-2-67-82.

ISSN 2782-2001, <http://journals.nstu.ru/vestnik>
Analysis and data processing systems
 Vol. 82, No 2, 2021, pp. 67–82