

УДК 004.048

***СРАВНИТЕЛЬНЫЙ АНАЛИЗ АКТИВАЦИОННЫХ ФУНКЦИЙ В ЗАДАЧЕ
ПРОГНОЗИРОВАНИЯ ВРЕМЕННЫХ РЯДОВ***

Науменко В.А.

Магистр

Национальный Исследовательский Ядерный Университет «МИФИ»

Москва, Россия

Аннотация: данная статья посвящена сравнению функций активации во входном слое глубокой нейронной сети и их влиянию на результаты прогноза будущих значений курса Bitcoin. Подобные сравнения прежде не проводились с глубокими нейронными сетями, использующими в качестве данных для обучения и прогнозирования курсы криптовалют. В рамках работы приведено описание характерных свойств активационных функций. Для разработки и обучения нейросети применялись библиотеки языка Python: pandas, numpy, matplotlib, keras, tensorflow.

Ключевые слова: глубокие нейронные сети, активационная функция, криптовалюта, прогнозирование, машинное обучение, библиотека Keras

***COMPARATIVE ANALYSIS OF ACTIVATION FUNCTIONS IN THE
PROBLEM OF FORECASTING TIME SERIES***

Naumenko V.A.

Master

National Research Nuclear University “MEPhI”

Moscow, Russia

Annotation: this article is devoted to comparing activation functions in the deep neural input layer and their influence on the results of forecasting future Bitcoin exchange rate values. Such comparisons don't have been made before with deep neural networks that use cryptocurrency exchange rates as data for training and forecasting. The article describes the characteristic properties of activation functions. To develop and train the neural network, Python libraries were used: pandas, numpy, matplotlib, keras, tensorflow.

Keywords: deep neural networks, activation function, cryptocurrency, forecasting, machine learning, Keras library

Прогнозирование динамики цен различных котировок всегда представляло сложную задачу для аналитиков и трейдеров. Знание будущих тенденций гарантирует инвесторам прибыль, в которой они заинтересованы. Постоянно разрабатываются системы, оказывающие помощь трейдерам и аналитиками в анализе ситуации на рынке, обрабатывая огромные потоки данных с учетом постоянно меняющейся ситуации.

Кроме того, существуют традиционные инструменты, такие как технический анализ, фундаментальный анализ, анализ объемов. Фундаментальный анализ подразумевает всестороннее изучение компаний и компаний – «лидеров» в этой области, их экономические показатели, сопоставление с конкурентами. Технический анализ включает в себя сравнение исторических данных, представленных в виде графиков, с графическими паттернами, на основе, которых можно предположить дальнейшее поведение анализируемого графика. Но эти методы не являются адаптивными, они не могут предоставить полной картины происходящего. В отличие от нейронных сетей с их богатым функционалом. В частности, удачно составленная структура нейронной сети с определенными параметрами способна с большой долей вероятности заметить то, что традиционные методы не способны определить. Один из таких параметров может оказаться активационная функция. В настоящее время для решения задачи прогнозирования временных рядов все

чаще прибегают к использованию методов глубокого обучения. Это связано с тем, что они представляют лучшие результаты предсказания нежели методы машинного обучения. Нейронные сети относятся к семейству моделей глубокого обучения [4]. С течением времени возрастает их популярность как финансовых инструментов.

Первоначально сверточные нейронные сети предназначались для анализа изображений, для их классификации [5-7]. Для этих целей использовался набор фильтров, через которые проходило изображение, при этом применялись весовые коэффициенты к разным аспектам изображения. Таким образом сеть учится на изображениях и отбирает те фильтры, которые влияют на процесс классификации. Аналогичный подход может быть применен и к временным рядам. В данном случае сверточная сеть выступает как скользящее окно, позволяющее отобрать у хаотичного временного ряда особые признаки, которые непосредственным образом оказывает влияние на его динамику с течением времени [3].

Данная работа посвящена сравнительному анализу активационных функций для уже составленной нейронной сети, реализованной с помощью библиотеки Keras, строящей прогноз для котировок криптовалюты Bitcoin.

Данные для анализа были взяты с сайта компании Финам за период с 1 января 2016 по 1 января 2020 года с периодичностью 5 минут [9].

На сегодняшний день существует различные активационные функции. В частности, в библиотеке Keras реализованы ReLU function, sigmoid function, softmax function, softplus function softsign function, tanh function (гиперболический тангенс), selu function, elu function, exponential function. Стоит обратить внимание на самые популярные среди них [11]. Функция ReLU на данный момент является самой используемой в связи с ее значительными преимуществами: быстрым подсчетом производной и разреженностью активации, то есть более

оптимизированное использование нейронов при их большом количестве. В то же время у этой функции есть недостаток, сказывающийся на ее эффективности [2]. Это прекращение обучения при градиентном спуске в следствие неизменности весов на участках с нулевой производной. Это обстоятельство может вынудить пересмотреть целесообразность использования данной функции в угоду более подходящих для конкретной задачи. Помимо функции ReLU, есть также гиперболический тангенс и softmax. Гиперболический тангенс применяется для получения быстрой сходимости, а softmax удобен для моделей с малым количеством слоев [8].

Для нейросети, составленной последовательно из слоев Conv1D, Flatten и двух Dense (рис.1), в слое Conv1D в activation_function подставлялись функции ReLU, гиперболический тангенс и softmax. В каждом отдельном случае происходило обучение сети с выбранной функцией активации и строился прогноз на проверочной выборке (рис.2-7). Построение графиков осуществлялось в Google Colaboratory [1,10]. Архитектура сети и графики, построенные на ее основе, являются собственной разработкой.

```
model = Sequential()  
model.add(Conv1D(50, 5, input_shape = (xTrain.shape[1], xTrain.shape[2]), activation_function))  
model.add(Flatten())  
model.add(Dense(10, activation="linear"))  
model.add(Dense(yTrain.shape[1], activation="linear"))  
model.compile(loss="mae", optimizer=Adam(lr=1e-4))  
model.fit(xTrain, yTrain, epochs=20, batch_size=20, verbose=1, validation_data=(xVal, yVal))
```

Рис. 1 – Разработка сети с использованием библиотеки Keras в Python

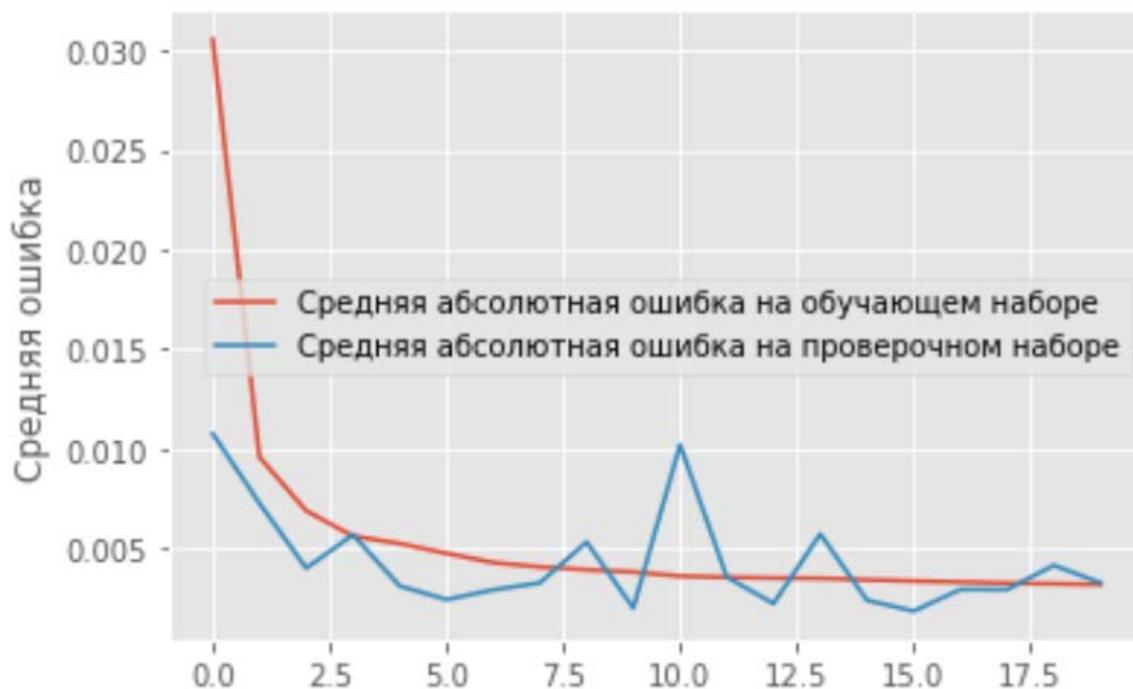


Рис. 2 – Средняя абсолютная ошибка на обучающей и проверочной выборках для активационной функции ReLU

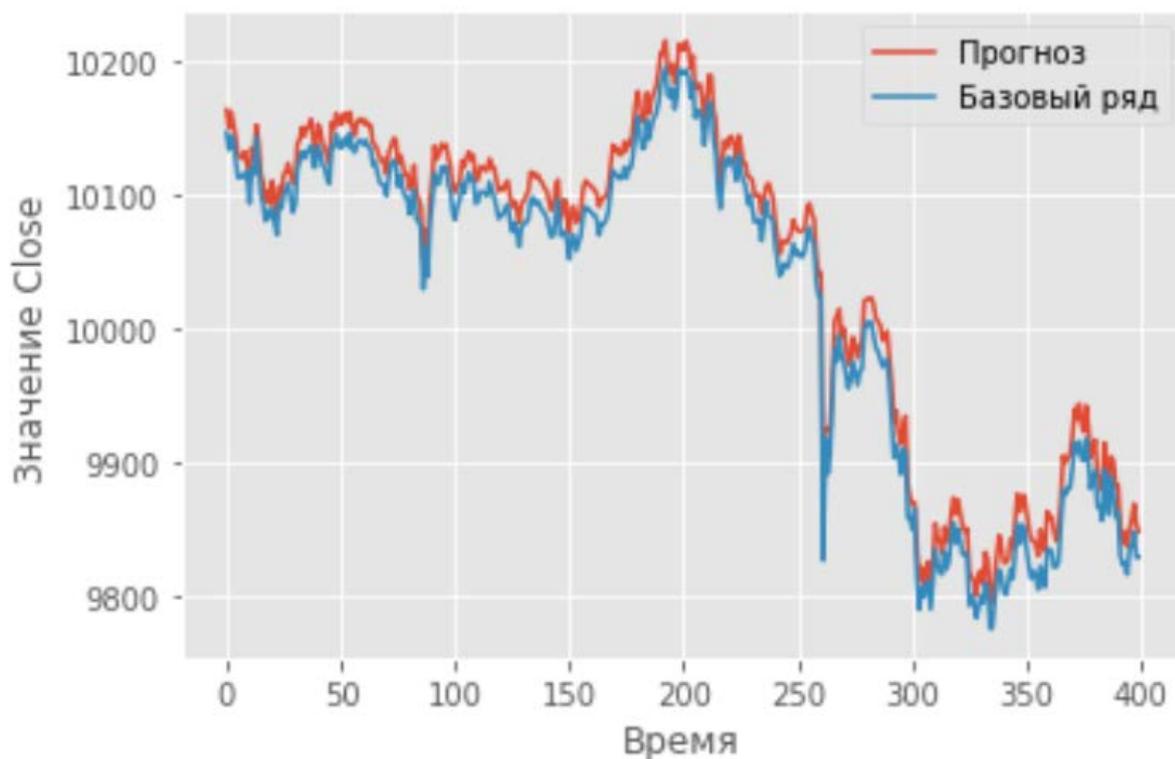


Рис. 3 – Прогноз на проверочной выборке для активационной функции ReLU



Рис. 4 – Средняя абсолютная ошибка на обучающей и проверочной выборках для активационной функции гиперболический тангенс

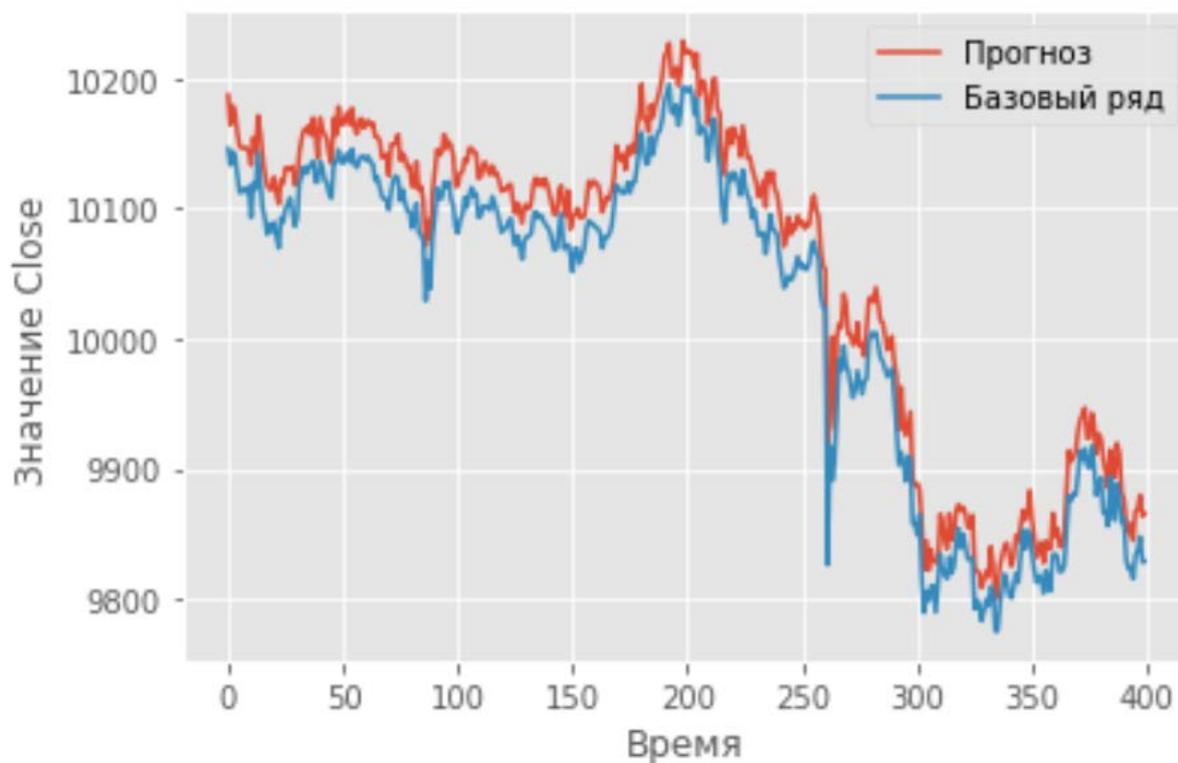


Рис. 5 – Прогноз на проверочной выборке для активационной функции гиперболический тангенс



Рис. 6 – Средняя абсолютная ошибка на обучающей и проверочной выборках для активационной функции softmax

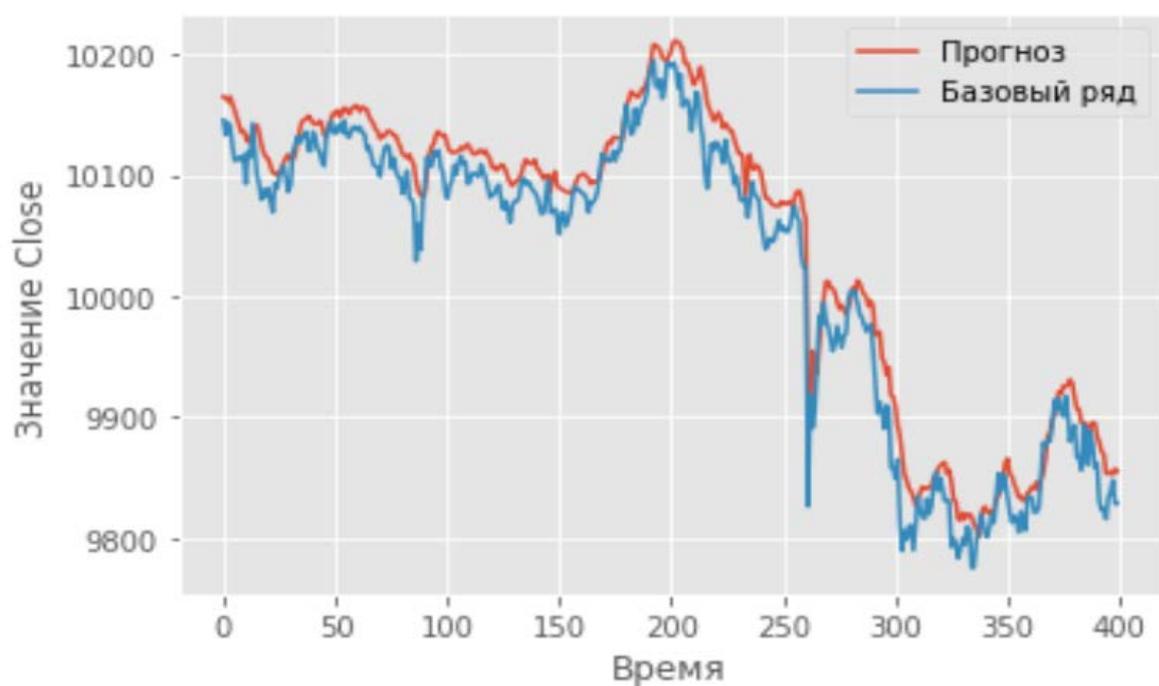


Рис. 7 – Прогноз на проверочной выборке для активационной функции softmax

На основе полученных графиков можно сделать вывод о том, что из рассматриваемых функций ReLU дает наименьшую ошибку на тестовой выборке и вследствие этого прогноз получается довольно точным. Гиперболический тангенс также предоставляет схожее поведение графика прогноза с базовым рядом хоть и с большей погрешностью по сравнению с ReLU. Чего не скажешь о функции softmax, которая позволяет получить лишь повторяет по контуру основные пики базового ряда. Он более сглаженный по сравнению с другими прогнозами. Результаты данной работы могут быть использованы биржевыми аналитиками и трейдерами при работе с реальными данными. Кроме того, архитектура сети представляет интерес исследователям, работающим с машинным обучением.

Библиографический список

1. Майк МакГрат. Программирование на Python для начинающих. (перевод с английского М. А. Райтмон) Москва. – Эксмо 2015. – 192с.
2. Николенко, Р. Глубокое обучение нейронных сетей / Р. Николенко, Р. Ульянова. – М.: Техносфера, 2005. – 1072 с.
3. Татьянкин В.М., Дюбко И.С. Нейронные сети глубокого доверия в сравнении с многослойным персептроном // Вестник Югорского государственного университета. 2015. Вып. 2(37). С.87-89.
4. Хайкин С. Нейронные сети: полный курс, 2-е издание.: Пер. с англ. – М.: Издательский дом «Вильямс», 2006. 1104с.
5. Deng L. Deep Learning: Methods and Applications [Текст] / Deng L., Yu D. // Foundations and Trends in Signal Processing Vol. 7. - 2014. - P. 197-387.
6. He K., Zhang X., Ren S., et al. Deep Residual Learning for Image Recognition // 2016 IEEE Conference on Computer Vision and Pattern Recognition (Las Vegas, NV, USA, 27-30 June 2016), 2016. P. 770778. DOI: 10.1109/CVPR.2016.90

7. LeCun Y., Bengio Y., Hinton G. Deep Learning // Nature. 2015. Vol. 521. P. 436-444. DOI: 10.1038/nature14539
8. Rav'i D., Wong Ch., Deligianni F., et al. Deep Learning for Health Informatics //IEEE Journal of Biomedical and Health Informatics. 2017. Vol. 21, No. 1. P. 4-21. DOI: 10.1109/JBHI.2016.2636665
9. Котировки на мировых биржах онлайн / [Электронный ресурс]. – Режим доступа: <https://www.finam.ru/quotes> (дата обращения 20.04.2020).
10. Google Colaboratory / [Электронный ресурс]. – Режим доступа: <https://colab.research.google.com/> (дата обращения 12.05.2020).
11. Keras: The Python Deep Learning library / [Электронный ресурс]. – Режим доступа: <https://keras.io/> (дата обращения 20.04.2020).

Оригинальность 89%