

Применение методов машинного обучения для анализа показателей на финансовых рынках

¹***КЛЮЕВА Елена Георгиевна**, старший преподаватель, e.klyueva@kstu.kz,
¹**СОЛОДОВНИКОВА Ирина Валентиновна**, старший преподаватель, irinasolo@mail.ru,
²**КОТЛЯРОВ Николай Александрович**, магистрант, nikolaykotlyarov@gmail.com,
¹НАО «Карагандинский технический университет имени Абылкаса Сагинова»,
 пр. Н. Назарбаева, 56, Караганда, Казахстан,
²Казахстанско-Британский технический университет, ул. Толе би, 59, Алматы, Казахстан,
 *автор-корреспондент.

Аннотация. Представлены результаты исследований в области применения различных методов машинного обучения (ML) для анализа и прогнозирования временных финансовых рядов. Цель исследования – провести оценку пригодности алгоритмов ML для решения задачи анализа и прогнозирования состояния финансовых рынков. В качестве исходных данных использовались открытые данные об акциях Apple за период с мая 2002 по июнь 2020 года. В рамках исследования были разработаны алгоритмы машинного обучения с применением стохастического анализа и произведена подготовка данных с последующим предпроцессингом для лучшего обучения алгоритмов, как показано на примере формирования матрицы для алгоритма бустинга. Была изучена область прогнозирования финансовых временных рядов и применены алгоритмы бустинга, полносвязанной нейронной сети, рекуррентной нейронной сети, нейронных сетей с длинной кратковременной памятью, конволюционной нейронной сети. На основе обученных алгоритмов был проведен сравнительный анализ на выборке финансовых инструментов из одного сектора.

Ключевые слова: машинное обучение, нейронные сети, анализ данных, бустинг, полносвязанная нейронная сеть, рекуррентная нейронная сеть, нейронная сеть с длинной кратковременной памятью, конволюционная нейронная сеть, web-приложение, финансовые показатели.

Введение

Прогнозирование хаотических временных рядов – задача, которая давно стоит перед различными областями науки и бизнеса. Точное прогнозирование временных рядов на много шагов вперед может дать бизнесу серьезное конкурентное преимущество.

В математике временной ряд – это набор данных, измеряемых последовательно через определенные промежутки времени. Прогнозирование – это попытка использовать настоящие и прошлые данные для прогнозирования этих данных на будущее. Сегодня данные накапливаются в огромных количествах, и человечество обладает значительными вычислительными мощностями для обработки накопленных данных. Это позволяет ученым и инженерам решать проблему предсказания хаотических временных рядов и делать прогнозы за достаточно короткое

время.

Целью настоящего исследования является разработка системы, использующей алгоритмы нейронных сетей для учета более сложных нелинейных зависимостей актива и краткосрочного прогнозирования будущего состояния финансового временного ряда.

Материалы и методы исследования

Машинное обучение в финансах сегодня считается ключевым аспектом ряда финансовых услуг и приложений, включая управление активами, оценку уровня риска, расчет кредитных рейтингов и даже одобрение кредитов.

Общим подходом к моделированию финансовых временных рядов является использование стохастического процесса дискретного времени [1]. Из-за сложности и неинтуитивного характера большинства

моделей машинного обучения пользователь может не понять, как именно работает модель, а следование за решениями через очень сложный и большой набор математических шагов является неразумным и часто невыполнимо для любого реального приложения. Вместо этого идеальным вариантом является наличие объяснимой модели машинного обучения. Объясняемая модель – это модель, которая не является понятной по своей сути, но результаты которой можно объяснить с помощью методов *post-hoc*. Благодаря более объяснимому ИИ, тщательному выбору алгоритмов и обработке данных можно выяснить, почему принимаются решения, почему все идет не так, и, соответственно, скорректировать стратегии для поддержания прибыльности. В данном исследовании используются современный алгоритм *boosting trees* – CatBoost [2]. Он поддается объяснению *post-hoc*, т.е. SHAP, эффективен и легко настраивается. На текущий момент нет исследований, изучающих, применимы ли SHAP-объяснения CatBoost, которые получены из аппроксимации оригинальной модели, к финансовым рынкам.

Нейронные сети – это алгоритмы, созданные по образцу человеческого мозга и предназначенные для распознавания закономерностей. Они используют форму машинного зондирования для интерпретации, маркировки или группировки необработанных данных. Недавно предложенная Эндрю Ыном [3] новая вычислительная архитектура для нейронных сетей названа графическим блоком (GU). GU – это простая рекуррентная сеть нейронов, каждый из которых оснащен отдельным аттрактором, соответствующим одному значению входа. Каждый нейрон получает входные данные от набора других нейронов, называемого локальным входным множеством. Локальные входные множества набора нейронов образуют входные множества GU. Входы нейрона изменяются с помощью ряда преобразований, соответствующих количеству входящих нейронов. Нейроны могут работать циклически, по очереди. Когда нейрон срабатывает, он может сработать снова, посылая импульс нейронам в своем локальном входном наборе. Сила импульса пропорциональна количеству нейронов в локальном входном наборе. GU тесно связана с рядом ранее предложенных нейронных сетей, включая популярные сети Хопфилда [4].

Рекуррентная нейронная сеть (RNN) – это особый тип искусственной нейронной сети, которая может обрабатывать данные в виде временных рядов или последовательностей [5]. Благодаря своей внутренней памяти, RNN могут запоминать важные вещи о

полученных данных, что позволяет им очень точно предсказывать, что произойдет дальше. Рекуррентные нейронные сети могут развивать гораздо более глубокое понимание последовательности и ее контекста по сравнению с другими алгоритмами.

RNN особенно хорошо подходят для обработки последовательных данных, поскольку наличие контуров обратной связи позволяет им использовать как текущие, так и прошлые входные данные, что позволяет сохранять информацию. Это свойство RNN означает, что они могут обучаться и учитывать тенденции и контекст при обучении и составлении прогнозов. Однако существует существенное ограничение – они теряют память в долгосрочной перспективе из-за проблемы исчезновения градиента. Для решения этой проблемы Хохрайтер и Шмидхубер в 1997 году представили сеть долговременной памяти (LSTM) [6].

Сверточная или конволюционная нейронная сеть (ConvNet/CNN) – это алгоритм глубокого обучения, который может принимать входное изображение, присваивать важность (обучаемые веса и сдвиги) различным аспектам/предметам изображения и отличать один от другого. Для ConvNet требуется гораздо меньше предварительной обработки, чем для других алгоритмов классификации. В то время как в примитивных методах фильтры разрабатываются вручную, ConvNets могут обучиться этим фильтрам/функциям при надлежащем обучении [5]. Операция свертки направлена на извлечение высокоуровневых характеристик.

В рамках исследования было принято решение использовать следующие алгоритмы машинного обучения: метод бустинга, полностью связную нейронную сеть, рекуррентную нейронную сеть, нейронную сеть с длинной кратковременной памятью, конволюционную нейронную сеть.

Разрабатываемую систему было решено реализовать в виде web-приложения, используя следующие проектные решения:

- для построения моделей и тренировки моделей был выбран язык программирования Python;

- для обучения моделей был выбран фреймворк Pytorch;

- для построения графиков была выбрана библиотека matplotlib;

- для разработки функционала стохастических моделей был выбран пакет Scikit;

- для загрузки данных (временных рядов) была выбрана библиотека yahoo finance;

- для манипуляций с данными был выбран фреймворк pandas.

Реализация метода бустинга

Бустинг – композиционный метаалгоритм

машинного обучения, применяется, главным образом, для уменьшения смещения (погрешности оценки), а также дисперсии в обучении с учителем [7].

Реализация бустинга на алгоритмах линейной регрессии была построена на данных об акциях Apple с интервалом 1 час, полученных из общедоступных источников [8]. По данным можно сделать вывод, что графики Open, High, Low, Close (распределение показателей цены на момент открытия, закрытия для часового интервала, также распределения наибольшей и наименьшей цены) схожи и, следовательно, вредны для линейных моделей и бустинга, так как данные коллинеарны (рисунок 1). Поэтому для обучения будет взята 1 колонка данных из 4 коллинеарных.

Далее создается матрица из временного ряда таким образом, чтобы в векторе ответов был один элемент (значение цены в конкретное время). Далее дизъюнктивно разделим матрицу на 2. На первой будет проводиться этап обучения, на второй оценим качество алгоритма.

После этого применяется метод бустинга на подготовленных данных и получим следующие оценки: на тестовой выборке точность составила 0.93 и на тренировочных 0.94, что показано на рисунке 2. Метрикой являлась точность совпадения реальной цены акции с предсказанной. Алгоритм сначала обучился на подготовленной матрице для обучения, итеративно улучшая метрику, затем проведена оценка алгоритма на тестовой матрице.

Реализация полносвязанной нейронной сети

Для целей обучения и тестирования данные по каждой акции были разделены на блоки по 750 дней для обучения (примерно три года торговли), 270 дней для оценки (более одного года торговли) и 270 дней для тестирования (рисунок 3).

Класс нейронной сети [5], который тренировал и проверял сеть, был построен следующим образом:

```
class NeuralNetwork:
```

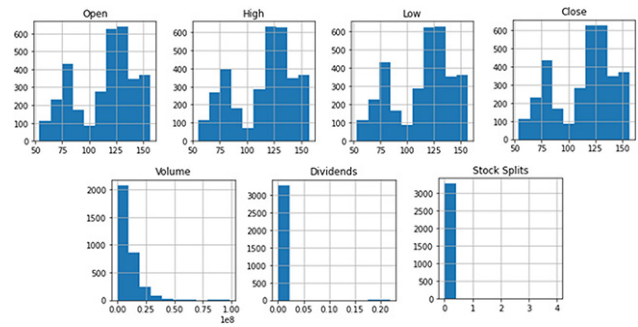


Рисунок 1 – Данные об акциях Apple на 5 минутном интервале

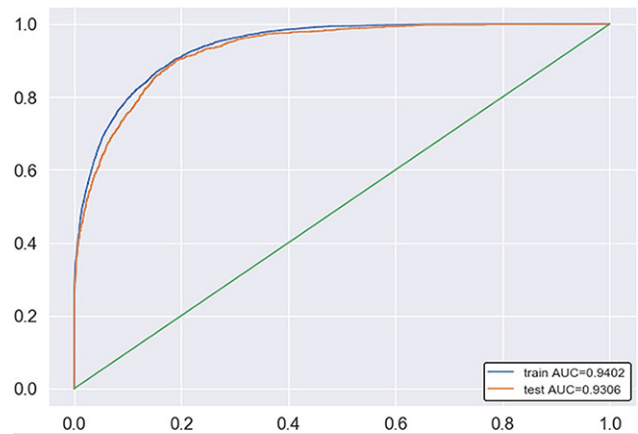


Рисунок 2 – Метрики

```
def __init__(self, x, y):
    self.input = x
    self.weights1 = np.random.rand(self.input.
    shape[1], 4)
    self.weights2 = np.random.rand(4, 1)
    self.y = y
    self.output = np.zeros(self.y.shape)
def feedforward(self):
    self.layer1 = sigmoid(np.dot(self.input,
    self.weights1))
    self.output = sigmoid(np.dot(self.layer1,
    self.weights2))
def backprop(self):
    # application of the chain rule to find
    derivative of the loss function with respect to
    weights2 and weights1
```

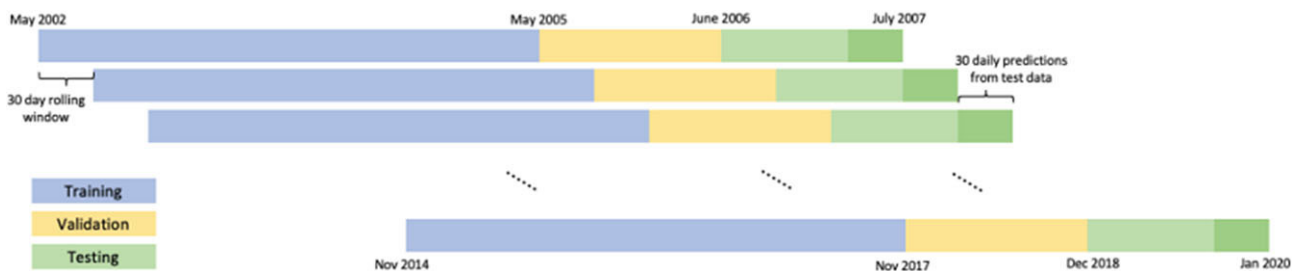


Рисунок 3 – Разделение данных

```

d_weights2 = np.dot(self.layer1.T, (2 *
(self.y - self.output) * sigmoid_derivative(self.
output)))
d_weights1 = np.dot(self.input.T, (np.
dot(2 * (self.y - self.output) * sigmoid_
derivative(self.output),
self.weights2.T) * sigmoid_derivative(self.
layer1)))
# update the weights with the derivative
(slope) of the loss function
self.weights1 += d_weights1
self.weights2 += d_weights2

```

После тренировки на данных получился результат с минимальными ошибками, как показано на рисунке 4. С каждым проходом по подготовленным данным количество неверных предсказаний уменьшалось (с 2000 до 270).

Реализация рекуррентной нейронной сети

Для реализации сети экспериментальным путем была выбрана архитектура, представленная на рисунке 5.

Считается, что использование комбинации моделей нейронных сетей уменьшает изменчивость и улучшает обобщение [5]. Поэтому, как и в предыдущем случае, вместо одной RNN были использованы несколько независимых RNN, обученных параллельно.

Поскольку RNN являются независимыми, их выходы также независимы. Окончательный совместный прогноз того, будет ли доходность следующего дня выше или ниже медианы, основывается на минимальном количестве LSTM, называемом порогом. Другими словами, порог – это минимальное количество RNN, которые должны соответствовать прогнозу. Поскольку существует 11 RNN, любой порог не менее шести считается большинством.

Чтобы изучить влияние различных пороговых значений и требований к минимальной точности на среднедневную доходность, эти ковариаты RNN варьировались. Результат при изменении порога. Если исключить три случая, которые кажутся аномальными, то можно увидеть, что среднедневная доходность портфеля растет по мере увеличения порога, пока он не достигнет восьми, после чего доходность начинает снижаться. Слишком низкий порог означает, что в портфель могут быть включены даже акции, прогнозируемые хуже медианы. С другой стороны, после оптимального числа восемь, условия становятся слишком жесткими и требуют большего количества ROC для достижения того же прогноза, в результате чего в портфель добавляется меньше акций или не добавляется вообще, как показано на рисунке 6.

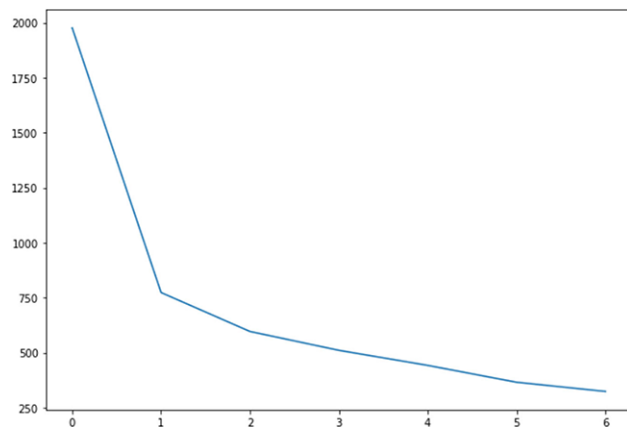


Рисунок 4 – Результат выполнения

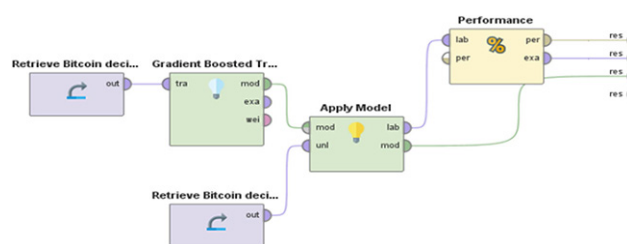


Рисунок 5 – Архитектура рекуррентной нейронной сети

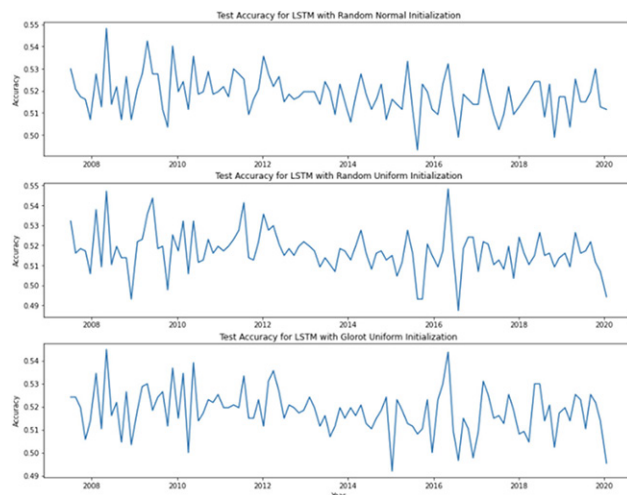


Рисунок 6 – Точность модели

Реализация нейронных сетей с длинной кратковременной памятью

Для реализации нейронной сети с длинной кратковременной памятью экспериментальным путем была выбрана архитектура, представленная на рисунке 7.

В торговой стратегии акции, которые, по прогнозам LSTM, превзойдут ожидания, покупаются и добавляются в портфель с тем же весом. Акции удерживаются до тех пор, пока модель не перестанет предсказывать,

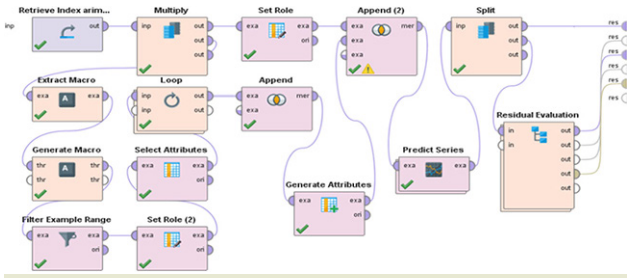


Рисунок 7 – Архитектура нейронной сети с длинной кратковременной памятью

что акции превзойдут медиану, после чего позиция закрывается. Полученный портфель является довольно динамичным и ежедневно корректируется на основе прогнозов LSTM. Портфель не содержит фиксированного количества акций; он содержит столько акций, сколько, по прогнозам модели, даст хорошие результаты.

Модель LSTM состоит из входного нейрона, скрытого слоя и выходного нейрона. На выходе используется сигмоидная функция активации, которую можно интерпретировать как меру доверия. Значение ближе к 1 означает, что модель более уверена, что доходность будет выше медианы, а значение ближе к 0 означает, что она более уверена, что доходность будет ниже медианы. Использовался метод оптимизации Адама вместе с коэффициентом обучения 0,0075, который был выбран с помощью байесовской оптимизации.

Полученные результаты представлены на рисунке 8.

Реализация конволюционной нейронной сети

Для реализации сети экспериментальным путем была выбрана архитектура, показанная на рисунке 9.

CNN на основе внимания (ACNN) способен улавливать глобальные и локальные зависимости, чего не может сделать LSTM, что повышает надежность [6]. Структура ACNN-LSTM может быть использована как часть структуры для кодирования и декодирования. В когнитивной системе человека внимание обычно предшествует памяти. ACNN способен улавливать долгосрочную зависимость, потому что он сочетает в себе внимание с множественными головами и сверткой. Комбинация LSTM и ACNN может улучшить как структурные преимущества, так и способность моделирования временных рядов [9]. Благодаря интеграции многоголового внимания и многомасштабного сверточно-го ядра, ACNN может лучше улавливать сложность, чего не может сделать LSTM, в то время как LSTM может лучше представлять

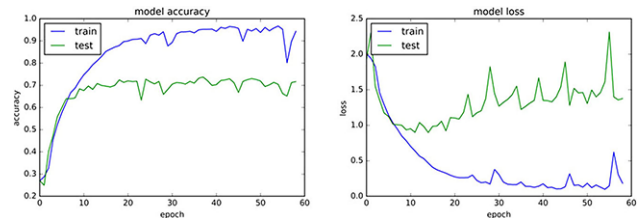


Рисунок 8 – Результат работы нейронной сети с длинной кратковременной памятью

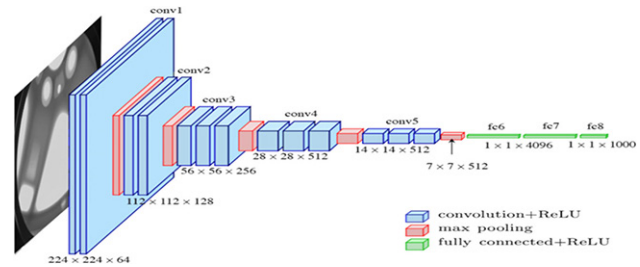


Рисунок 9 – Архитектура конволюционной нейронной сети

свойства временного ряда.

Были получены результаты, представленные на рисунке 10.

Результаты и их обсуждение

В ходе данного исследования был проведен анализ различных алгоритмов машинного обучения. Были использованы реальные данные и проведено обучение алгоритмов.

Алгоритм полностью связанной нейронной сети показал себя как алгоритм, не точно показывающий тенденцию временного ряда, и не справился с задачей, как видно на графике, красная линия далеко от эталонной синей (рисунок 11).

Алгоритм рекуррентной нейронной сети показал себя как алгоритм, сильно зависящий от начальных значений нейронов, при больших значениях временной ряд будет предсказываться очень выше реальных показателей, обратная ситуация, когда начальные веса малы и приводят к большому разбросу алгоритма, что делает его неэффективным (рисунок 12).

Алгоритм нейронной сети с длинной кратковременной памятью показал себя как алгоритм, весьма точно предсказывающий показатель временного ряда, красная линия прогнозируемого результата находится очень близко с реальными значениями (рисунок 13).

Алгоритм конволюционной нейронной сети показал себя как неэффективный для данной задачи, прогнозирование алгоритма

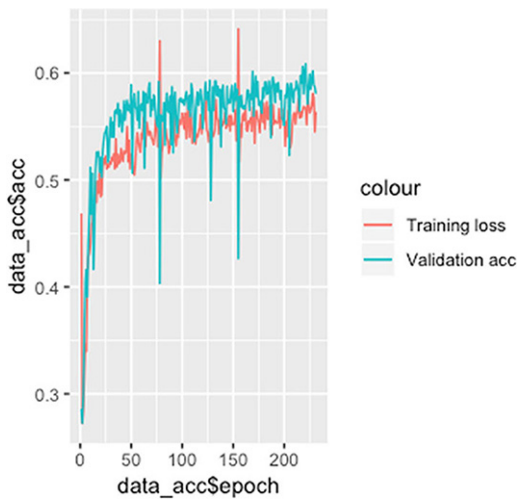


Рисунок 10 – Результаты работы конволюционной нейронной сети



Рисунок 11 – Предсказанные результаты алгоритма полносвязанной нейронной сети



Рисунок 12 – Предсказанные результаты алгоритма рекуррентной нейронной сети

неточное (рисунок 14).

Алгоритм бустинга показал себя как алгоритм, способный прогнозировать основную тенденцию временного ряда, но с отклонением вверх или вниз, отработал хорошо, но хуже, чем алгоритм нейронной сети с длиной кратковременной памятью (рисунок 15).

Заключение

Стохастическое моделирование факторных систем связей между аспектами экономической деятельности опирается на обобщение закономерностей изменения значений экономических показателей – количественных характеристик факторов и



Рисунок 13 – Предсказанные результаты алгоритма нейронной сети с длиной кратковременной памятью

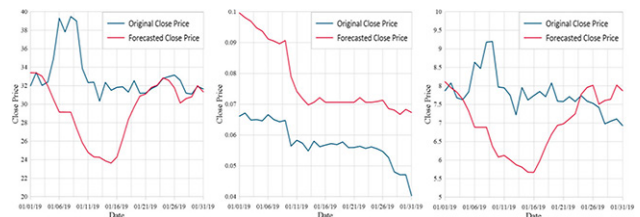


Рисунок 14 – Предсказанные результаты алгоритма конволюционной нейронной сети



Рисунок 15 – Предсказанные результаты

экономических показателей. Количественные параметры взаимосвязи определяются путем сравнения значений исследуемых показателей по совокупности экономических объектов или временных периодов.

В рамках данного исследования был проведен сравнительный анализ разработанных и обученных алгоритмов ML на выборке финансовых инструментов из одного сектора, оценено качество моделей как по индивидуальным метрикам, так и на результатах реальных данных.

Разработанная система имеет базу данных с финансовыми инструментами на разных временных интервалах, показывает графики обучения и все метрики, релевантные для алгоритма, есть возможность распараллеливания вычислений и вычислений на видеокарте.

Данная работа является применимой в области предиктивной аналитики, в любом исследовании, где фигурируют временные ряды, также возможно расширение видов алгоритмов.

СПИСОК ЛИТЕРАТУРЫ

1. Вентцель, Е.С. Теория случайных процессов и ее инженерные приложения / Е.С. Вентцель, Л.А. Овчаров – М.: Мнемозина, 2000. – 251 с.
2. Hancock, J.T., Khoshgoftaar, T.M. CatBoost for big data: an interdisciplinary review. J Big Data 7, 94 (2020). <https://doi.org/10.1186/s40537-020-00369-8>.
3. Andrew, Ng Machine Learning Yearning / Ng Andrew. – Текст: электронный // nessie.ilab.sztaki.hu: [сайт]. – URL: https://nessie.ilab.sztaki.hu/~kornai/2020/AdvancedMachineLearning/Ng_MachineLearningYearning.pdf (дата обращения: 05.09.2022).
4. Cao J.D., Tao Q. Estimation on domain of attraction and convergence rate of Hopfield continuous feedback neural networks // Journal of Computer and System Sciences, 2001. 62. Pp. 528-534.
5. Golovko, V. Neural Networks and Artificial Intelligence / V. Golovko, A. Imada. – Springer, 2014. – Vol. 440. Communication in Computer and Information Science. – 270 p.
6. Sak H., Senior A., Beaufays F. Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling // INTERSPEECH – 2014. – Singapore: ISC, 2014. – Pp. 338-342.
7. Friedman J., Hastie T., Tibshirani R. Additive Logistic Regression: a Statistical View of Boosting. – 1998.
8. Datasets. – Текст: электронный // Kaggle: [сайт]. – URL: <https://www.kaggle.com/datasets> (дата обращения: 06.09.2022).
9. A. Graves and J. Schmidhuber // Framewise phoneme classification with bidirectional LSTM and other neural network architectures. Neural Networks, 18(5):602-610, 2005. IJCNN 2019.

Қаржы нарығындағы көрсеткіштерді талдау үшін машиналық оқыту әдістерін қолдану

¹***КЛЮЕВА Елена Георгиевна**, аға оқытушы, e.klyueva@kstu.kz,

¹**СОЛОДОВНИКОВА Ирина Валентиновна**, аға оқытушы, irinasolo@mail.ru,

²**КОТЛЯРОВ Николай Александрович**, магистрант, nikolaykotlyarov@gmail.com,

¹«Әбілқас Сағынов атындағы Қарағанды техникалық университеті» КеАҚ, Н. Назарбаев даңғылы, 56, Қарағанды, Қазақстан,

²Қазақстан-Британ техникалық университеті, Төле би көшесі, 59, Алматы, Қазақстан,

*автор-корреспондент.

Аңдатпа. Қаржылық уақыттық қатарларды талдау және болжау үшін әртүрлі машиналық оқыту әдістерін (ML) қолдану саласындағы зерттеулердің нәтижелері берілген. Зерттеудің мақсаты – қаржылық нарықтардың жағдайын талдау және болжау мәселесін шешу үшін ML алгоритмдерінің жарамдылығын бағалау. Бастапқы деректер 2002 жылдың мамырынан 2020 жылдың маусымына дейінгі кезеңдегі Apple акциялары туралы жалпыға қолжетімді деректер болды. Зерттеу бөлігі ретінде машиналық оқыту алгоритмдері стохастикалық талдау арқылы әзірленді және күшейту алгоритмі үшін матрицаны қалыптастыру мысалында көрсетілгендей, жақсырақ оқыту алгоритмдері үшін кейіннен алдын ала өңдеу арқылы деректер дайындалды. Қаржылық уақыт қатарын болжау саласы зерттелді және күшейту алгоритмдері, толық қосылған нейрондық желі, қайталанатын нейрондық желі, ұзақ қысқа мерзімді жады бар нейрондық желілер, конволюционды нейрондық желі қолданылды. Оқытылған алгоритмдер негізінде бір сектордағы қаржы құралдарының іріктемесіне салыстырмалы талдау жүргізілді.

Кілт сөздер: машиналық оқыту, нейрондық желілер, деректерді талдау, күшейту, толық қосылған нейрондық желі, қайталанатын нейрондық желі, ұзақ қысқа мерзімді жады нейрондық желі, конволюционды нейрондық желі, веб қолданбасы, қаржылық өнімділік.

Application of Machine Learning Methods for the Analysis of Indicators in the Financial Markets

¹***KLYUYEVA Yelena**, Senior Lecturer, e.klyueva@kstu.kz,

¹**SOLODOVNIKOVA Irina**, Senior Lecturer, irinasolo@mail.ru,

²**KOTLYAROV Nikolay**, Master's Student, nikolaykotlyarov@gmail.com,

¹NPJSC «Abylkas Saginov Karaganda Technical University», N. Nazarbayev Avenue, 56, Karaganda, Kazakhstan,

²Kazakh-British Technical University, Tole Bi Street, 59, Almaty, Kazakhstan,

*corresponding author.

Abstract. The results of research in the field of application of various machine learning (ML) methods for the analysis and forecasting of financial time series are presented. The purpose of the study is to assess the suitability of ML algorithms for solving the problem of analyzing and predicting the state of financial markets. The source data was public data on Apple shares for the period from May 2002 to June 2020. As part of the study, machine learning algorithms were developed using stochastic analysis and data was prepared with subsequent preprocessing for better learning algorithms, as shown in the example of matrix formation for the boosting algorithm. The area of forecasting financial time series was studied and algorithms of boosting, fully connected neural network, recurrent neural network, neural networks with long short-term memory, convolutional neural network were applied. Based on the trained algorithms, a comparative analysis was carried out on a sample of financial instruments from one sector.

Keywords: machine learning, neural networks, data analysis, boosting, fully connected neural network, recurrent neural network, long short-term memory neural network, convolutional neural network, web application, financial performance.

REFERENCES

1. Ventcel', E.S. Teoriya sluchajnyh processov i ee inzhenernye prilozheniya / E.S. Ventcel', L.A. Ovcharov – Moscow: Mnemozina, 2000. – 251 p.
2. Hancock, J.T., Khoshgoftaar, T.M. CatBoost for big data: an interdisciplinary review. J Big Data 7, 94 (2020). <https://doi.org/10.1186/s40537-020-00369-8>.
3. Andrew, Ng Machine Learning Yearning / Ng Andrew. – Tekst: elektronnyj // nessie.ilab.sztaki.hu: [sajt]. – URL: https://nessie.ilab.sztaki.hu/~kornai/2020/AdvancedMachineLearning/Ng_MachineLearningYearning.pdf (data obrashcheniya: 05.09.2022).
4. Cao J.D., Tao Q. Estimation on domain of attraction and convergence rate of Hopfield continuous feedback neural networks // Journal of Computer and System Sciences, 2001. 62. Pp. 528-534.
5. Golovko, V. Neural Networks and Artificial Intelligence / V. Golovko, A. Imada. – Springer, 2014. – Vol. 440. Communication in Computer and Information Science. – 270 p.
6. Sak H., Senior A., Beaufays F. Long Short-Term Memory Recurrent Neural Network Architectures for Large Scale Acoustic Modeling // INTERSPEECH – 2014. – Singapore: ISC, 2014. – Pp. 338-342.
7. Friedman J., Hastie T., Tibshirani R. Additive Logistic Regression: a Statistical View of Boosting. – 1998.
8. Datasets. – Tekst: elektronnyj // Kaggle: [sajt]. – URL: <https://www.kaggle.com/datasets> (data obrashcheniya: 06.09.2022).
9. A. Graves and J. Schmidhuber // Framewise phoneme classification with bidirectional LSTM and other neural network architectures. Neural Networks, 18(5):602-610, 2005. IJCNN 2019.