Оглавление

[Вступление 2](#_Toc101842199)

[Технические индикаторы 7](#_Toc101842200)

[Модели 11](#_Toc101842201)

[ARIMA 11](#_Toc101842202)

[SVR 12](#_Toc101842203)

[Нейронные сети 14](#_Toc101842204)

[RNN 16](#_Toc101842205)

[LSTM 18](#_Toc101842206)

[Конфигурация модели 22](#_Toc101842207)

[Идентификация модели ARIMA 23](#_Toc101842208)

[Метрики качества моделей 24](#_Toc101842209)

[MSE 24](#_Toc101842210)

[RMSE 24](#_Toc101842211)

[R-квадрат 24](#_Toc101842212)

[MAE 25](#_Toc101842213)

[Оценка прибыльности 25](#_Toc101842214)

[Коэффициент Шарпа 26](#_Toc101842215)

[Анализ весов переменных 27](#_Toc101842216)

[Предсказания моделей на периоде валидации 28](#_Toc101842217)

[Результаты 31](#_Toc101842218)

[Прибыльность торговых стратегий на основе моделей 33](#_Toc101842219)

[Вывод 35](#_Toc101842220)

[Список литературы 37](#_Toc101842221)

Нужна более четкая структура. Меньше основных разделов. Четче разделить, где обзор литературы, где гипотезы и методология, отдельные разделы - данные, результаты, заключение. Нужно ввести нумерацию. В разделах могут быть подразделы, для них тоже нумерация, например 2.1. Сейчас не понятно, Технические индикаторы – это продолжение обзора литературы или выбранные в ВКР индикаторы?

# Введение

Машинное обучение и нейронные сети применяются в различных исследовательских областях; они доказали свою способность предсказывать процессы лучше, чем статистические методы. Достижения в области компьютерных наук позволяют проводить новые исследования в области экономики, это вызывает необходимость сравнивать и оценивать новые методы для конкретной задачи. Гибридные модели в настоящее время являются методом совершенствования моделей, поскольку они позволяют применять сложные методы без создания новых нейронных сетей. В то же время менее требовательные к вычислениям статистические методы и модели машинного обучения могут показать сопоставимые результаты в некоторых работах. Эти группы моделей необходимо изучить, чтобы определить их сильные и слабые стороны.

Постановка задачи. Учитывая то, что гибридные модели представляют интерес во многих областях и не изучены досконально, есть место для работы над гибридными моделями (лучше немного конкретнее). Это особенно важно, поскольку в некоторых случаях гибридные модели значительно превосходят исходные модели (ссылки).

Практическая значимость. Гибридные модели, сочетающие машинное обучение и модель ARIMA, уже используются для прогнозирования временных рядов в различных областях применения: для прогнозирования уровня воды (Phan & Nguyen, 2020), экспорта (Дейв и др., 2021), трафика (Ма и др., 2020), концентрации загрязняющих веществ (Ван и др., 2017). В последнее время гибридные модели применяются для прогнозирования процессов, связанных с COVID-19, таких как новые случаи заболевания (Мохан и др., 2022), госпитализации (Перон, 2021), доступность медицинских ресурсов (Кету и Мишра, 2021). Что касается прогнозирования фондового рынка с помощью гибридных моделей, эта область слабо исследована. Например, крайне мало работ, в которых исследуется гибридизация ARIMA с моделями машинного обучения на российском фондовом рынке и на развивающихся рынках в целом (ссылки). Это показывает необходимость дополнительных исследований в этой области, поскольку модели становятся все более актуальными в различных областях, но остаются неизученными при анализе финансовых временных рядов.

Цель данного исследования – прогнозирование \_\_\_\_\_ (чего именно? динамики цен акций?) с применением гибридных моделей на российском фондовом рынке. Для достижения этой цели необходимо создать модели ARIMA-LSTM и ARIMA-SVM и сверить их с наивными прогнозами, чтобы убедиться в их состоятельности. Сравнительный анализ разработанных моделей будет проводиться по различным метрикам, включая: \_\_\_\_\_\_\_\_\_\_\_ , что позволит сопоставить качество оценок согласно данным моделям как можно более всесторонне. Модели необходимо тестировать на низко- и высокочастотных данных, поскольку может возникнуть разница в производительности из-за перенастройки на небольших наборах данных. Кроме того, необходимо проверить, что внешние факторы улучшают прогнозную способность моделей, а не вызывают переобучение.

Задачи исследования:

1. Сбор оригинальной базы данных по рынкам .. ., включая показатели … за … годы
2. Построение и тестирование прогнозной способности моделей ….
3. …

Гипотезы (нужно переписать более четко)

1. Гибридные модели будут работать лучше, чем отдельные модели, с точки зрения выбранных показателей (каких именно?).
2. Стратегии “Купи и продай” с гибридными моделями будут значительно более прибыльными, чем соответствующие стратегии “Купи и держи” для всех моделей.
3. Гибридная модель с нейронной сетью LSTM покажет наилучшую предсказательную способность на основе всех или большинства выбранных показателей (каких?).
4. Включение внешних факторов (каких? Макропоказателей?) улучшает качество моделей на основе всех или большинства выбранных показателей (каких?).

Ограничения исследования. В исследовании основное внимание уделяется более простым гибридным моделям, избегая гибридов нескольких моделей машинного обучения, поскольку трудно определить подходящую комбинацию. Предмет данной работы ограничен российским фондовым рынком. Исследование не включает анализ смоделированных временных рядов, как это делают некоторые работы.

Обзор литературы

Гибридизация моделей. Статистические методы неэффективны для полного отражения реальных процессов, которые представляют собой линейные и нелинейные закономерности. Тем не менее, их способность моделировать линейные процессы может способствовать созданию более сложных методов. В одном из первых соответствующих исследований сравнивались отдельные модели SVM и ARIMA с их комбинацией, которая оказалась наиболее успешной (Pai & Lin, 2005). Это происходит потому, что разнородные модели могут снизить риск выбора неподходящей модели и позволяют моделям корректировать друг друга. Хашей и Хиджази (2008) заявили об этом в своей статье, в которой они протестировали несколько нейронных сетей, а также их комбинации с ARIMA на трех популярных наборах данных реального мира из разных областей. Они обнаружили, что во всех них гибриды показали лучшие результаты. В более поздней работе (Li et al., 2019), в которой рассматривалась проблема прогнозирования динамики (?) фондового рынка, сравнивались модели LSTM, SVM с их гибридом с ARIMA. Авторы обнаружили, что объединение моделей улучшает их способности к прогнозированию (Li et al., 2019). Они использовали высокочастотные и низкочастотные данные, чтобы определить, есть ли разница в производительности (что понимаем под производительностью? Качество прогноза? Нужно уточнить). То же самое будет сделано и в настоящей работе, поскольку меньшее количество выборок может негативно повлиять на производительность нейронной сети, в то время как SVM не требует большого объема данных.

Метрики качества. При оценке производительности нейронных сетей исследователи чаще всего используют базовые показатели, такие как MSE, RMSE и MAE. Существуют документы, которые добавляют к их оценке показатели прибыли и риска. Ханссон (2017) и Бек и др. (2018) включили как прибыльность стратегии “купи и продай”, так и коэффициент Шарпа. Другой распространенной метрикой является коэффициент R, который измеряет, насколько хорошо модель объясняет данные. При этом изучаются все вышеперечисленные показатели, чтобы охватить различные аспекты производительности и иметь возможность сравнивать результаты с большим количеством моделей.

Выбор моделей. ARIMA - самая популярная статистическая модель для прогнозирования временных рядов (ссылки). Его способность улавливать линейные закономерности отлично подходит для балансировки нелинейных моделей машинного обучения. Первая модель, объединенная с ARIMA, это SVR, которая представляет собой регрессионную версию широко используемой машины опорных векторов. Он оказался очень хорош в обобщении и нахождении нелинейных закономерностей (ссылки). Эти качества делают SVR очень непохожим на ARIMA, поэтому он является лучшим кандидатом для гибридизации. Другая модель - LSTM, представляющая собой рекуррентную нейронную сеть, которая была создана специально для анализа временных рядов и доказала свою высокую адаптивность (Петерсен и др., 2019). Сезер и др. (2020) обнаружили, что RNN используются в 51% всех работ, в которых исследуется применение нейронных сетей для прогнозирования финансовых временных рядов. Они также показали, что LSTM был наиболее распространенным типом RNN. Таким образом, LSTM является хорошим кандидатом для объединения с ARIMA, поскольку это одна из самых надежных моделей в этой области. Более того, исследование гибридизации LSTM очень важно, потому что более продвинутые и прибыльные модели в статьях являются гибридами LSTM. Например, Бао и др. (2017) представили WSAES-LSTM, который представляет собой комбинацию трех моделей: вейвлет-преобразования (WT), автоэнкодеры с накоплением (SAEs) и долговременная кратковременная память (LSTM). Их новая модель обеспечивает в среднем примерно в 3 раза более высокие показатели прибыльности (по сравнению с чем) и большую стабильность в этих оценках на 6 финансовых рынках. Кроме того, исследование показывает, что отбор активов в портфель на основе нейронных сетей могут достигать сопоставимых уровней прибыльности в развивающихся и развитых странах. Таким образом, исследование этих моделей на российском рынке является полезным и важным.

Чаще всего модели объединяются путем обучения остатков одной модели к другой. Существуют и другие методы объединения прогнозов, такие как голосование, усреднение и ансамблевые модели. Они плохо работают на разных моделях (? На каких моделях? Нужно точнее, дать ссылки), так как нет одновременного распознавания нескольких паттернов.

Первым шагом к выбранному процессу является определение параметров и коэффициентов модели ARIMA с использованием методологии Бокса-Дженкинса. Затем необходимо провести диагностическую проверку оценочных остатков, чтобы убедиться, что модель подходит для этих данных. После этого выбранная модель машинного обучения подгоняется к вычисленным остаткам. Наконец, остатки двух моделей объединяются для получения окончательной модели.

Данные. Технические индикаторы используются в большинстве моделей LSTM в литературе главным образом потому, что они почти всегда улучшают способность нейронных сетей к прогнозированию. Есть несколько теорий о том, почему это происходит. Например, в одном исследовании утверждается, что технические индикаторы могут помочь извлечь общедоступную информацию о фундаментальной стоимости ценных бумаг (Трейнор, 1985). Большинство исследований, в которых используется LSTM для прогнозирования фондового рынка, включают от 8 до 12 показателей (Сезер и др., 2020). Кара представила набор из 10 технических индикаторов, который часто использовался в более поздних исследованиях (Кара, 2011). В настоящем исследовании будет использоваться этот набор (какой? Как у Кара?), поскольку он доказал свою эффективность в моем предыдущем исследовании (если оно не опубликовано, ссылаться некорректно. Можно ссылаться на препринты, working paper, статьи).

Цена ценной бумаги не может быть объяснена только ее прошлыми значениями, существуют различные факторы, влияющие на цену. Анатольев (2008) показал растущую зависимость российского фондового рынка от фондового рынка США. Пересецкий (2011) использовал ARIMA и GARCH, чтобы определить, что существует влияние фондового рынка Японии и цены на нефть марки WTI. Лозинская и Жемчужников (2017) использовали все вышеупомянутые факторы, 1-месячную ставку Московской межбанковской оферты и индекс DAX. В статье сравнивалась нейронная сеть, обучаемая с использованием многослойного персептрона (MPL) и SVM. Примечательно, что SVM показал лучшие результаты, чем MLP, по всем показателям в работе, возможно, из-за способности SVM к обобщению и склонности нейронной сети к переобучению. Это особенно важно, потому что SVM требует значительно меньших вычислительных возможностей для обучения. Эти факторы могут способствовать переоснащению модели, поэтому необходимо провести тестирование улучшенных моделей путем добавления этих факторов. В данном исследовании (в ВКР?) в качестве внешних факторов выбраны индексы немецкой, японской и французской бирж (а почему США не рассматривается? Такой вопрос обязательно зададут, если речь о ВКР, также как и про Японию – почему не Китай), а также цена на нефть марки Brent и курс доллара.

Не хватает таблиц с систематизацией результатов прошлых работ. Работа – тип модели – выборка - решаемая задача – результат

Обычно не менее 5 стр занимает обзор. Выделяется 4-5 ключевых работ, по ним подробное описание дается (на развернутый абзац по каждой)

# Технические индикаторы

Технические индикаторы строятся на основе прошлых цен и объема торгов для предсказания цен, имеется множество теоретических и эмпирических подтверждений их предсказательной способности в литературе (ссылки). Теоретическое объяснение этому сводится, как правило, к неэффективности финансового рынка. Однако, предложено несколько моделей, по-разному объясняющих данную взаимосвязь. Первый вид модели предполагает, что при помощи технических индикаторов можно извлечь непубличную информацию о фундаментальной стоимости ценной бумаги (Treynor, 1985). Второй вид модели не утверждает наличие связи между техническими индикаторами и фундаментальной стоимостью, но говорит о том, что технические индикаторы формируют ожидания инвесторов и, как следствие, спекулятивное поведение, не связанное с фундаментальным анализом (Giovanni, 2011). Третий вид модели заключается в том, что сделки, отражающие положительные ожидания инвесторов, могут вызывать чрезмерные или, наоборот, недостаточные реакции, что создает паттерны, которые технические индикаторы могут выявить (Hong, 1999, Soros, 2015). Четвертый тип модели относится к настроению инвесторов, исследователи пришли к выводу, что настроения инвесторов могут объяснить премию за риск, а технические индикаторы успешно предсказывают изменения в настроениях инвесторов (Keynes, 1936, Neely, 2014).

Большинство работ, которые предсказывают котировки по LSTM-модели, включают так же технические индикаторы. Причем выбор количества технических индикаторов широко варьируется. Ниже приведены несколько работ, выпущенные за последние 5 лет, где моделью для предсказания являлась LSTM.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Номер работы | Работа | Набор данных и изучаемый рынок | Период | Тип данных | Набор показателей |
| 1 | (Agrawal et al., 2019) | 3 акции на индийском рынке | 2016 | Дневные | OHLCV и 2 ТИ |
| 2 | (Chen et al., 2015) | 3049 акций на китайском рынке | 1990–2015 | Дневные | OHLCV |
| 3 | (Dezsi, 2016) | 1 акция на румынском рынке | 2001–2016 | Дневные | OHLCV |
| 4 | (Borovkova, 2015) | 22 акций крупных компаний на американском рынке | 2017 | Пятиминутные | OHLCV и 12 ТИ |
| 5 | (Chen et al., 2018) | 1 фьючерс на китайский индекс CSI 300 | 2017 | Минутные | OHLCV и 1 ТИ |
| 6 | (Nelson et al., 2017) | 5 акций на бразильском рынке | 2008–2015 | Пятнадцати-минутные | OHLCV и 175 ТИ |
| 7 | (Zhou et al., 2018) | 42 акции на китайском рынке | 2016 | Минутные | OHLCV и 9 ТИ |
| 8 | (Tan et al., 2020) | 1 фьючерс на китайском рынке | 2016 | Получасовые | OHLCV и 21 ТИ |
| 9 | (Khare et al., 2017) | 10 акций на Нью-Йоркской бирже | 1 год | Минутные | Цены закрытия и 8 ТИ |

Таблица 1. Примеры работ по применению моделей глубокого обучения для предсказания финансовых временных рядов.

Как видно, число включенных технических индикаторов не связано с количеством выбранных бумаг или типом данных. В основном же исследователи ограничиваются использованием 12 индикаторами. Довольно много работ ограничиваются только OHLCV-переменными, но, если технические индикаторы все-таки добавляются к набору данных, их число, как правило, составляет от 9 до 12.

Кроме того, существует несколько видов технических индикаторов: индикаторы тренда, индикаторы скорости изменения, осцилляторы (ссылки). Индикаторы тренда используются для выявления тренда, могут быть неэффективны, когда цена акции на протяжение долгого времени находится внутри канала, то есть колеблется в одном интервале.

Другой тип индикаторов – индикаторы скорости изменения или индикаторы моментума – указывает на скорость изменения цены бумаги. На основе данного индикатора можно судить об изменении тренда. Кроме того, данный тип индикаторов лучше интерпретируется совместно с индикаторами тренда, так их пересечение говорит об изменении тренда, то есть совместное использование данных индикаторов в модели может значительно улучшить ее. Осцилляторы, третий тип индикаторов, показывают на отклонение цены от среднего значения. Они указывают на приближение коррекции или направление фазы колебания цены. Осцилляторы полезны для анализа в периоды отсутствия выраженных трендов.

Одним из самых часто используемых в литературе набором технических индикаторов является набор из работы «Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange» (Kara, 2011). Набор состоит из 10 технических индикаторов, представленных ниже.

|  |  |  |
| --- | --- | --- |
| Индикаторы тренда | Индикаторы объема | Индикаторы моментум |
| Simple Moving Average | Accumulation/Distribution Oscillator | Relative Strength Index |
| Linear Weighted Moving Average |  | The William’s %R oscillator |
| Commodity Channel Index |  | Momentum |
| Moving Average Convergence/Divergence |  | Stochastic %K |
|  |  | Stochastic %D |

Таблица 2. Набор построенных индикаторов и их типы.

Формулы для расчета технических индикаторов представлены ниже.

1. Простое скользящее среднее (Simple Moving Average) является среднем значением цены за определенный период:

где это цена закрытия дня t, n – временное окно индикатора.

1. Линейно-взвешенное скользящее среднее (Linear Weighted Moving Average) отличается от простого скользящего среднего тем, что веса предыдущих цен линейно уменьшаются:

где это цена закрытия дня t, n – временное окно индикатора.

1. Индикатор моментума (Momentum) измеряет величину изменения цены за определенный период:

где это цена закрытия дня t, n – временное окно индикатора.

1. Схождение/Расхождение Скользящих Средних (Moving Average Convergence/Divergence) показывает текущий тренд цены. Индикатор является соотношением двух скользящих средних и считается по формуле ниже:

где

1. Индекс Товарного Канала (Commodity Channel Index) измеряет отклонение цены инструмента от его среднестатистической цены, то есть занижена или завышена ли цена на основе скользящей средней:

где и равно простому скользящему среднему значений за n минут в соответствии с временным окном.

1. Индекс Относительной Силы (Relative Strength Index) это следующий за ценой осциллятор, который колеблется в диапазоне от 0 до 100, определяет продолжит ли цена двигаться по существующему тренду:

где и вычисляются по тому же принципу, что и при расчете Индекса Среднего Направления Движения (ADMI).

1. Процентный диапазон Вильямса (Williams’ Percent Range, Williams’ %R) определять состояния перекупленности или перепроданности акции, а также дающий сигналы о развороте рынка:

где – самое высокое значение максимума за n дней, – самое низкое значение минимума за n дней.

1. Accumulation/Distribution Oscillator (A/D) оценивает, насколько силен текущий тренд:

где это цены закрытия, максимума и минимума дня t соответственно, n – временное окно индикатора.

1. Stochastic %K – это индикатор импульса, который показывает переоценена или недооценена ценная бумага:

где это цена закрытия дня t, n – временное окно индикатора, – самое высокое значение максимума за n дней, – самое низкое значение минимума за n дней.

1. Stochastic %D указывает на разворот рынка:

Так как на высокочастотных данных значения часто повторяются, может возникнуть много значений NaN или бесконечностей. Nan – это результат вычисления, который не может быть представлен как число. Бесконечность может получиться при делении на ноль. Это является серьезной проблемой, так как модели глубокого обучения не способны обучаться, даже если есть одно такое значение. В случае создания NaN-значений при построении индикаторов к ним применена линейная интерполяция. Данный метод заменят пропущенные значения на средние арифметические ближайших валидных значений.

Используются только индикаторы как таковые? А сигналы, которые они дают, применяются? Если применяются, то нужен список таких сигналов

# Модели

## ARIMA

ARIMA-модель является одной из самых популярных статистических линейных моделей (ссылки). Классическая модель используется для предсказания одномерных временных рядов при помощи декомпозиции ряда на текущие, прошлые значения и случайные ошибки. Модель ARIMA(p, d, q) является комбинацией следующих моделей:

1. AR(p) – авторегрессионная модель порядка p, которая моделирует временной ряд как аддитивную линейную функцию p прошлых значений ряда согласно формуле ниже:

где – константная величина, – процесс белого шума, – коэффициенты авторегрессии.

1. MA(q) – модель скользящего среднего q-го порядка, представляющая линейный ряд через аддитивную линейную функцию q прошлых ошибок согласно формуле ниже:

где – процесс белого шума, – коэффициенты скользящего среднего.

Параметр d определяет порядок дифференцирования, необходимый для достижения стационарности ряда. Дифференцирование необходимо для финансовых временных рядов, так как в литературе доказано, что они не являются стационарными (Visser, 2010). Данный факт делает ARMA-модель неприменимой для предсказания цен финансовых инструментов. Нестационарные временные ряды можно привести к стационарному виду при помощи взятия разности порядка d. Интегрированный временной ряд можно представить в следующем виде:

Таким образом, модель ARIMA(p, d, q) может быть представлена как:

где – процесс белого шума, – параметры модели, – оператор разности порядка d.

Построение ARIMA-модели по методологии Бокса-Дженкинса состоит из следующих шагов[[1]](#footnote-1):

1. Идентификация модели.
2. Оценивание коэффициентов модели
3. Тестирование и диагностика модели.
4. Прогнозирование

## SVR

Метод опорных векторов (SVM, Support Vector Machine) основан на статистической теории обучения и был впервые представлен в работе «The Nature of Statistical Learning Theory» (Vapnik, 1995). Основная идея метода опорных векторов заключается в разделении объектов в многомерном пространстве оптимальным способом при помощи гиперплоскости. Оптимальной гиперплоскостью считается та, при которой расстояние между гиперплоскостью и объектами является максимальной.

Для решения задач регрессии в 1997 году была предложена модель Support Vector Regression (Vapnik et al., 1997), которая обобщает проблему классификации, предсказывая действительное число. Целью SVR-модели является определение вектор-функции , которая аппроксимирует все пары с точностью . Функцию так же стремятся сделать как можно более плоской, чтобы не допустить переобучение и сделать более устойчивой к выбросам. Математическое описание модели и логику ее построения в некоторых работах основывают на книге «Neural networks and learning machines» (Haykin, 2009)

В случае линейной функции , она может представлена в виде:

где – вектор весов, – отклонение, – входной вектор. Плоскость для функции (1) подразумевает маленькие веса , то есть необходима минимизация евклидовой нормы . Математически это условие можно записать как оптимизацию выпуклой функции:

Решение данной задачи существует, только если существует такая функция , которая аппроксимирует все пары с точностью Требуется введение дополнительных переменных, которые ослабляют требования для функции *.* Введем - независимые положительные переменные, описывающие, насколько функция потерь чувствительна к ошибкам – . Новое определение проблемы представлено ниже:

где C – гиперпараметр, который определяет обмен между плоскостью функцией и насколько отклонение может превышать Данная задача решается при помощи квадратичного динамического программирования.

## Нейронные сети

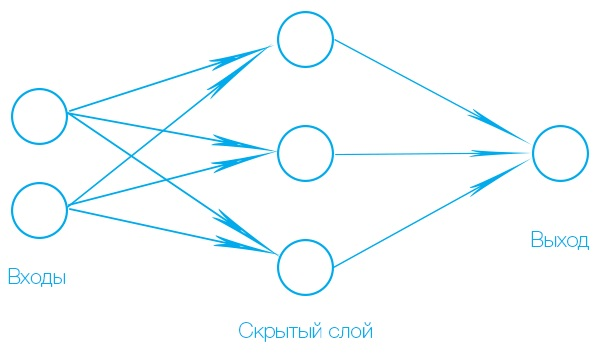


Рисунок 11. Архитектура простой нейронной трехслойной сети. Источник: Modelling and recognition of images [Электронный ресурс] // Простейшие нейронные сети на Python в Visual Studio URL: <https://api-2d3d-cad.com/python-neuron-vs/> (дата обращения: 15.04.21).

Базовое представление нейронной сети с тремя слоями. Например, для задачи анализа финансовых временных рядов Слой 1 (входной слой) состоит из набора независимых OCHLV-переменных. Слой 2 (один скрытый слой) представляет собой атрибуты первого слоя, которые не выражены в нем напрямую. Например, из High и Low модель может извлечь волатильность котировок в данную минуту. Стрелки из входного слоя в скрытый иллюстрируют создание неявных переменных из нескольких явных. Слой 3 (выходной слой) объединяет информацию из скрытых слоев и выдает предсказанное направление движения цены акции. Каждый круг в сети представляет собой нейрон, у которого тип совпадает со слоем, в котором он находится. Другими словами, нейронная сеть состоит из входных, скрытых и выходных нейронов. Обучение нейронной сети представляет собой поиск оптимальных весов, которые объединяют каждый слой нейронов. Основные метода поиска гиперпараметров это метод обратного распространения ошибки, генетический алгоритм и нечеткие алгоритмы.

Глубокое обучение работает следующим образом. Один из основных и самых мощных методов обучение является метод обратного распространения ошибки. На части тренировочных данных, после создания случайных матриц весов считается функция потерь, на основе ошибок формируется градиенты потерь. Наконец, с помощью градиентного спуска устанавливаются новые веса.

Существует две главные категории нейронных сетей: сверточные (CNN) и реккурентные (RNN). Как правило, они применяются для разных задач, но оба типа исследованы в литературе, и их качество сопоставлено. В работе «STOCK PRICE PREDICTION USING LSTM, RNN AND CNN-SLIDING WINDOW MODEL» средний процент ошибки реккурентной нейронной сети составил 5,217%, сверточной – 4,98% (Hiransha, 2017).

В то время в работе сверточных сетей слои соединяются между собой функциями, в рекуррентных сетях функциями так же соединяются прошлые значения переменной.

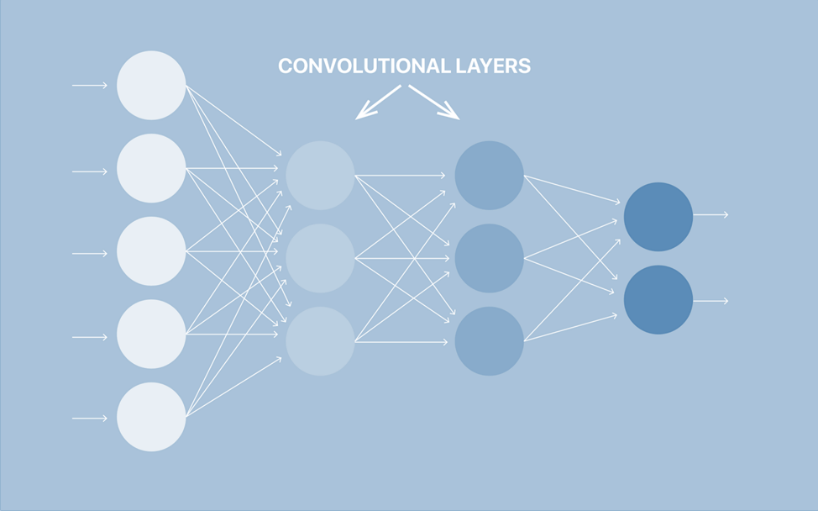


Рисунок 11. Архитектура сверточной нейронный сети. Источник: Lionbridge [Электронный ресурс] // What is the Difference Between CNN and RNN? URL: <https://lionbridge.ai/articles/difference-between-cnn-and-rnn/> (дата обращения: 15.04.21).

CNN отличны от других нейронных сетей наличием сверточных слоев, внутри которых информация трансформируется, после чего передается следующему слою. Трансформация происходит при помощи фильтров, которыми называют матрицы, которые меняют информацию на каждом слое. Данные сети плохо подходят для обработки последовательной информации такой как текст, видео, временные ряды, их применяют для классификации и распознавания объектов и лиц на фотографиях. Во-первых, в них не учитываются предыдущие значения. Во-вторых, они не устроены для изучения информации разной размерности. Тем не менее такой тип нейронных сетей все же используется для предсказания цен акций, но для этой задачи их построение достаточно усложнено. Показатели конвертируются в изображение или создается много сложенных слоев сети (в идеале нужны ссылки на статьи, монографии…).

## RNN

Рекуррентные сети созданы специально для обработки последовательной информации, их особенностью является способность нейронов получать информацию от самих себя предыдущего прохода. Под последовательной информацией понимается следующий набор данных:

где t является временным интервалом в периоде от 1 до (Goodfellow et al., 2016). В данном исследовании последовательностью является период с августа по декабрь 2020 года с интервалом в одну минуту.

Изображение выглядит как текст, доска

Автоматически созданное описание

Рисунок 12. Архитектура рекуррентной нейронный сети. Источник: Towards Data Science [Электронный ресурс] // Recurrent Neural Networks (RNNs) URL: <https://towardsdatascience.com/recurrent-neural-networks-rnns-3f06d7653a85> (дата обращения: 15.04.21).

На Рисунке 12 представлена визуализация рекуррентной сети в сложенном и разложенном виде. Разложенный вид представляет собой сеть из полной последовательности. Источник: Goodfellow I., Bengio Y., Courville A. (2016). Deep learning. MIT Press.

Например, для исследования одного часа, подобная сеть состояла бы из 60 слоев. Входное значение – x(t) – является входным значением нейронной сете на шаге t. В данном случае x(1) является числовым вектором состоящим из OHLCV-показателей и соответствующих технических индикаторов для первой минуты из последовательности. Нейрон из скрытого слоя – h(t) – представляет собой нейрон на шаге t, который выполняет функцию памяти о прошлом состоянии. Рассчитывается такая функция по следующей формуле:

где функция f является функцией активации. Функция активации определяет выходное значение нейрона. По формуле 3 видно, что выходное значение зависит от входного x(t) и от прошлого выходного значения . Так же функция зависит от матриц весов U и W, первая связывает входные и скрытые значения, вторая связывает скрытые значения с прошлыми скрытыми значениями.

Наконец, o(t) на изображении означает выходное значение нейрона. Оно так же связано матрицей с остальными переменными в сети. Матрица весов, соединяющая скрытый слой и выходное значение обозначается V. Эта матрица весов, как и остальные, неизменчива для каждого элемента. Такая связь матрицы и выходного значения представляется в уравнении (4). Изначально матрицы U, W и V генерируется из случайных распределений. Следующим шагом будет передача вперед (Forward pass), которая описывается системой уравнений ниже:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | (2) | | |
|  | | (3) | | |
|  | | (4) | | |
|  | | (5) | | |
| Где – нормализованные с помощью операции *softmax* вероятности выходных значений, b и c – смещения, изначально сгенерированные из нулей. В данном примере выбрана функция активации tahn, кроме нее так же могут быть выбраны: сигмоида, ReLu, softmax или другие. Как правило, выбор функции активации происходит при помощи перебора и определения, какая функция дает наибольшее значение. Система уравнений (2)-(5) определяет рекуррентную нейронную сеть с одинаковой длинной входных и выходных значений. Для следующего шага обучения необходимо определить функцию потерь. Функция может быть написано специально под определенную задачу, но одной из универсальных функций потерь является кросс-энтропия, прописанная ниже:   |  |  | | --- | --- | |  | (6) | |  | (7) |   Далее на основе данной функции потерь нужно пересчитать матрицы весов U, W, и V. Для этого, согласно формулам, ниже считаются градиенты функции потерь для данных матриц:   |  |  | | --- | --- | |  | (8) | |  |  | |  | (9) |   Аналогично уравнению (9) считается градиент функций потерь матрицы U. На данном моменте нужно обратить внимание на недостаток RNN для решения задачи предсказания цены акций. На Рисунке 13 представлены производные сигмоидной функции и гиперболического тангенса. Как видно, производные стремятся к нулю при увеличении модуля аргумента. Поэтому с каждым шагом происходит потухание, и не поддерживается тот же темп обучения. Другие функции такие как ReLu наоборот вызывают «взрыв» градиента. Единственным решением данной проблемы является усложнение архитектуры и применение LSTM.  Sigmoid and Tanh Activation Functions · A.I. Dan  Рисунок 13. Графики сигмоидной функции и гиперболического тангенса и их производные. Источник: AI Dan [Электронный ресурс] // Sigmoid and Tanh Activation Functions URL <https://a-i-dan.github.io/math_nn> (дата обращения: 15.04.21).  В теории проблемы с обработкой долговременных зависимостей у RNN быть не должно. Человек может аккуратно подбирать параметры сети для решения искусственные задачи такого типа. К сожалению, на практике обучить RNN этим параметрам кажется невозможным. Эту проблему подробно исследовали Зепп Хохрайтер (Sepp Hochreiter, 1991) и Иошуа Бенджио (Yoshua Bengio) с соавторами (1994); они нашли неоспоримые причины, по которым это может быть сложно. LSTM Long Short-term memory (LSTM) – разновидность архитектуры нейронных сетей, которые отличаются способностью к обучению как долгосрочных так и краткосрочных зависимостей (Goodfellow et al., 2016). | | | |  |
| Рисунок 14. Архитектура нейрона LSTM. Источник: Bao, W., Yue, J. and Rao, Y. (2017) ‘A deep learning framework for financial time series using stacked autoencoders and long-short term memory’, PLoS ONE, 12(7), p. 1–24.    Рисунок 15. Повторяющийся модуль в сети RNN. Источник: Bao, W., Yue, J. and Rao, Y. (2017) ‘A deep learning framework for financial time series using stacked autoencoders and long-short term memory’, PLoS ONE, 12(7), p. 1–24.  В сравнении заметно, насколько RNN проще по архитектуре чем LSTM, хотя и схожую цепную структуру. По Рисункам 15 и 16 заметно, что в RNN повторяющийся модуль состоит из одного слоя, который обозначается на картинке прямоугольником, в LSTM уже 4 слоя. На Рисунке 16 кругами обозначаются поточечные операции такие как сложение векторов.    Рисунок 16. Повторяющийся модуль в сети LSTM. – входной вектор, – выходной вектор, – значение скрытого нейрона, – значение входного фильтра, – значение фильтра забывания, - значение выходного фильтра на шаге t. – вектор значений кандидатов на шаге t. Bao, W., Yue, J. and Rao, Y. (2017) ‘A deep learning framework for financial time series using stacked autoencoders and long-short term memory’, PLoS ONE, 12(7), p. 1–24.  Математически Рисунок 16 описывается следующим образом:   1. – входное значение вектора в модуль на шаге t. 2. это матрицы весов. 3. – смещения. 4. – значение скрытого нейрона на шаге t. 5. – значение фильтра забывания на шаге t, рассчитываемое по формуле ниже:  |  |  | | --- | --- | |  | (10) |   Вычисление входного фильтра является первым шагом в работе модуля LSTM. Задачей данного шага является определение, какую информацию сохранить, а от какой избавиться. На данном этапе входное значение шага и скрытое значение нейрона прошлого шага проходят через сигмоидальные слой, называемый «слоем фильтра забывания». В итоге для каждого числа из вектора памяти функция возвращает число от 0 до 1, где 0 соответствует решению полностью сохранить информацию, а 1 – полностью выбросить. Например, в данном случае если произошел разворот тренда, то информация о движении акции до разворота становится неактуальной. Тогда модель скорее всего избавится от большей части этих значений.   1. – значение входного фильтра на шаге t:  |  |  | | --- | --- | |  | (11) |   Следующий шаг позволяет решить, какую информацию хранить в ячейке на основе предыдущего шага. Процесс начинается с определения, какие значения ячейки нужно заменить на новые, при помощи второго сигмоидального слоя, который называется «слоем входного фильтра». Функция так же возвращает значения от 0 до 1, но 0 соответствует решению полностью забыть информацию с прошлого состояния, а 1 – полностью оставить.   1. – вектор значений кандидатов на шаге t:  |  |  | | --- | --- | |  | (12) |   Далее строится вектор значений-кандидатов при помощи tanh функции, которые могут быть добавлены в состояние ячейки. В примере с изменением тренда, желаемым результатом на данном этапе будет создание значений-кандидатов с новыми значениями направления тренда.   1. – вектор памяти на шаге t:  |  |  | | --- | --- | |  | (13) |   После определения, какую информацию стоит сохранить, а также создания новых значений для замены старых, наконец можно изменить старое состояние ячейки на новое . Уравнение (13) объединяет все установленные ранее решения о новом состоянии модели. Старое состояние умножается на слой фильтра забывания , чтобы сохранить только нужную информацию. Новые значения-кандидаты () умножены на слой входного фильтра (). Получается новое состояние ячейки . В данном примере это состояние, когда модель уже изменила направление тренда.   1. – значение выходного фильтра на шаге t:  |  |  | | --- | --- | |  | (14) |   На данном этапе сигмоидальный слой определяет, какая информация будет сохранена на выходе из ячейки.   1. – значение скрытого нейрона на шаге t:  |  |  | | --- | --- | |  | (15) |   Последним шагом вектор памяти проходит через tanh-слой, чтобы вывести значения от -1 до 1. Наконец, этот вектор умножается на значения выходного фильтра на шаге t. При изменении тренда желаемое выходное значение, которое запомнит ячейка памяти. | | | |  |
| Конфигурация модели Перед проверкой гипотезы необходимо сконфигурировать модель, чтобы он наилучшим образом подходила для решения поставленной задачи. Кроме того, выбираемые параметры должны соответствовать компутационным возможностям, что так же сделает ее менее долгой и более применимой для работы на высокочастотных данных. В Таблице 7 представлены параметры, а также их описание и роль в модели. | | |  | |
| Количество сложенных слоев LSTM, выбор зависит от сложности решаемой задачи. Модель дольше обучается при увеличении количества слоев. | num\_layers | | | |
| Вероятность того, что выход любого данного нейрона будет удален. Используется для недопущения переобучения. | dropout | | | |
| Количество обучающих примеров за одну итерацию. Чем меньше показатель, тем меньше памяти требуется и меньше времени на обучение, но менее точные оценки. | batch\_size | | | |
| Коэффициент скорости обучения, меняется от 0 до 1. При малых значениях модель обучается дольше, слишком большие значения могут привести к неоптимальному решению. | learning\_rate | | | |
| Затухание скорости обучения, понижение скорости обучения с итерациями. Нужно, чтобы модель была более стабильной. | learning\_rate\_decay | | | |
| Общее количество эпох, качество становится лучше с каждой эпохой, но занимает больше времени. | epoch | | | |
| Количество шагов в обучающей итерации. Выбор зависит от свойств данных, выбирается путем перебора. | n\_steps | | | |

Таблица 7. Конфигурация модели для всех экспериментов в Keras.

Другой важной частью конфигурации модели является выбор оптимизатора. В свою очередь оптимизатор изменяет установленные вручную параметры. Например, при работе с двумя переменными оптимизатор изучает ландшафт функции, чтобы определить какая скорость обучения лучше всего подходит, чтобы достичь глобальный минимум функции потерь, не перескочив ее слишком большим шагом, но и не устанавливает ее настолько маленькой, что модель никогда не достигнет оптимального значения. Как видно из Таблицы 8 выбор оптимизатора может значительно повлиять на качество модели: при оптимизаторе SGD модели accuracy составляет 0.621, а при RMSprop – 0.813.

Самым популярным оптимизатором на 2020 год является стохастический градиентный спуск (SGD). Градиент считается как сумма градиентов, вызванных каждым элементом обучения. Среднеквадратичное распространение (RMSProp), который поддерживает скорости обучения по каждому параметру. По этой причине оптимизатор хорошо работает на «зашумленных» данных, которыми являются котировки ценных бумаг. Адаптивная оценка момента (Adam) является комбинацией оптимизатора RMSProp с оптимизатором Momentum.

Не хватает гиперпараметров модели, в таблице на предыдущей странице нет конкретных значений

# Идентификация модели ARIMA

Модель идентифицируется путем перебора моделей ARIMA с максимальным лагом до 10. Лучшая модель определяется при помощи информационного критерия Акаике-Шварца.

Лучшей модификацией модели-ARIMA стала ARIMA (1,1,0). Остатки также не являются нормальными, согласно тесту Пирсона. На Рисунке 17 показано несколько графиков, описывающие остатки ARIMA (1,1,0). В целом, остатки несильно отличаются от нормального распределения за исключением нескольких выбросов, согласно графику в левом нижнем углу. Остатки выглядят стационарными за исключением небольших колебаний волатильности, что естественно с высоко волатильными рыночными активами. Статистика Дурбина-Ватсона остатков равна 1.13, что говорит о наличие автокорреляции, но не высокой, это подтверждает коррелограмма, которая показывает высокую значимость 1 лага и небольшую значимость последующих лагов.

Численные проверки на стационарность проводились? Например, с помощью теста Дики-Фуллера?

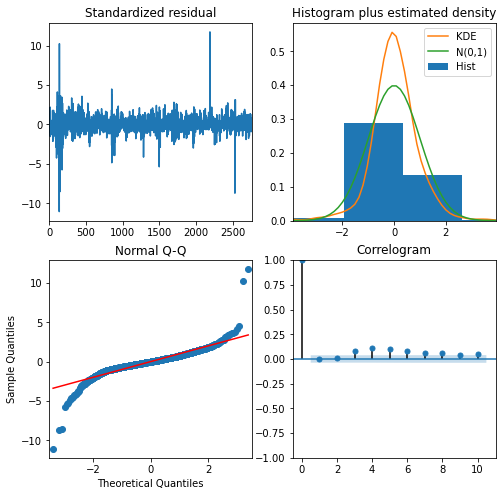


Рисунок 17. Графики остатков ARIMAX (1,1,0).

Метрики качества моделей

В данной работе будет использоваться несколько метрик качества: MSE[4,5,6,8,11], MAE[1,3,4,6,11], RMSE[1,7,11], R-квадрат[3,11], прибыльность по стратегии «buy-and-sell»[9,10] и коэффициент Шарпа[2,9,10]. Выбор критериев определяется частотой использования их в литературе по данной тематике, а также различиями в их интерпретации. Кроме того, выбор нескольких критериев позволяет сопоставить результаты с другими моделями.

MSE

Среднеквадратичное отклонение вычисляется по следующей формуле:

где - прогноз модели, - истинное значение. Среднеквадратичное отклонение является наиболее часто используемой метрикой в задачах регрессии в машинном обучении. Однако, данная величина плохо интерпретируется, так как измеряется в квадрате исходной величины. В связи с этим вводится ещё одна метрика – RMSE.

RMSE

Корень из среднеквадратичной ошибки сохраняет единицы измерения, и позволяет интерпретировать результаты моделей. Считается RMSE согласно формуле ниже:

где - прогноз модели, - истинное значение.

Обе метрики используют для сравнения модели между собой, но у MSE и RMSE есть существенный недостаток – метрики не позволяют сделать вывод о том, насколько хорошо модель решила задачу. Например, их невозможно даже грубо соотнести с моделями, которые решают задачу предсказания индексов другой стоимости.

R-квадрат

Коэффициент детерминации лежит в интервале от 0 до 1, что является метрикой того, насколько хорошо модель решила поставленную задачу. Данную метрику уже можно грубо соотносить с другими моделями, решающими такую же задачу. R-квадрат считается по формуле ниже:

где - прогноз модели, - истинное значение, – среднее значение целевой переменной. Коэффициент детерминации можно интерпретировать как нормированную среднеквадратичную ошибку. Он показывает долю дисперсии, объясненную моделью, в дисперсии целевой переменной. Соответственно, чем ближе показатель к нулю, тем лучше модель объясняет данные. Значение метрики, равное нулю, соответствует наивному предсказанию - – то есть константе.

MAE

Среднее абсолютное отклонение имеет следующую формулу:

где - прогноз модели, - истинное значение.

Модуль отклонения менее чувствителен к выбросам и лучше оценивает качество модели при большом количество выбросов.

Оценка прибыльности

Торговая стратегия «buy-and-sell» широко используется для оценивания прибыльности стратегии. Стратегия строится на уже предсказанных значениях каждой модели. Таким образом, модели оцениваются одной стратегией для определения наиболее прибыльной модели для инвестирования.

Согласно стратегии «buy-and-sell» покупка бумаги происходит, когда предсказанная цена в следующем периоде выше фактической цены в данном периоде. В противоположном случае, если предсказанная цена ниже фактической, бумага продается. Математическое представление данной стратегии представлено ниже:

где – фактическая стоимость в текущем периоде,- предсказанная цена следующего периода. Доходы стратегии определяются следующей формулой:

где R – доходность стратегии, b и c – общее число дней для покупки и продажи соответственно. B и C – транзакционная стоимость покупки и продажи.

Практическая имплементация данной стратегии затруднена высокими транзакционными издержками, связанными с торговлей акциями. Таким образом, стратегия применяется при торговле соответствующими фьючесными контрактами. В литературе доказана тесная связь индексом и соответствующих им фьючесов. В данном случае для корректности торговли фьючерсами вместо акций следует провести тест Спирмена на корреляцию и коинтеграцию, для подтверждения стабильной связи индекса и его фьючерса.

Кроме того, стратегию «buy-and-hold» можно использовать в качестве «бэнчмарка» для данного метода оценивания. Данная стратегия заключается в покупке индекса в начале рассматриваемого периода и продажи в конце периода. Если прибыльность стратегии «buy-and-sell» равна или ниже данной, то модель неприемлема.

Коэффициент Шарпа

где – кумулятивная доходность до периода t, и – среднее и стандартное отклонение доходности , – безрисковая процентная ставка. Коэффициент Шарпа отражает доходность, скорректированную на риск. В качестве безрисковой процентной ставки можно использовать значения кривой беспроцентной доходности для 3 месяцев, которые предоставляются Московской биржей. Ежедневные значения безрисковой процентной ставки в данном случае доступны с 2014 года.

Нужно сделать раздел, где описана выборка. Финансовые инструменты, годы, характеристики – среднее, медиана и пр. Указать, что является периодом обучения, что валидации

# Анализ весов переменных

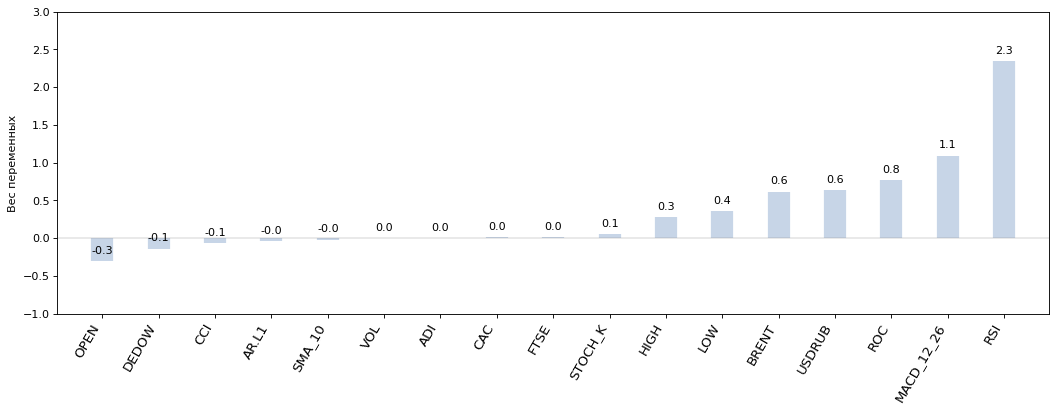


Рисунок 17. Веса в модели ARIMAX (1,1,0). Источник: расчеты автора.

**Мы сразу перешли к факторам и результатам. Нет раздела с гипотезами, кроме краткого их перечисления во введении. Нет обоснования гипотез на основе предыдущих работ. Нет обоснования выбора факторов**

Несмотря на то, что все переменные в модели значимы, многие из них имеют очень малый вес даже лаг переменной. Самыми важными для предсказания в данной модель являются технические индикаторы: RSI, MACD и ROC. Менее значимыми стали внешние переменные: курс доллара и цены нефти марки Brent.

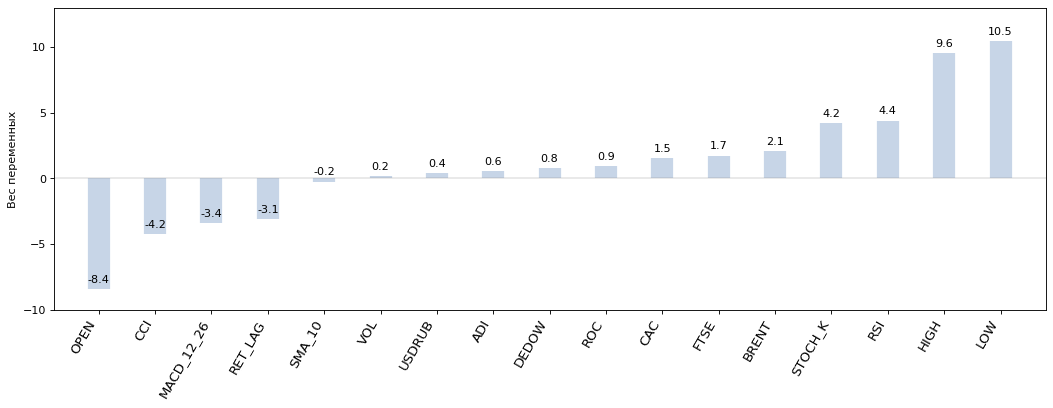


Рисунок 18. Веса в модели ARIMAX (1,1,0). Источник: расчеты автора.

Для SVR важными оказываются переменные-OHLV, уже после идут технические индикаторы – RSI, CCI, Stochastic K%.

Для LSTM невозможно посчитать веса переменных, что делает модель неудобной для интерпретации и анализа.

# Предсказания моделей на периоде валидации

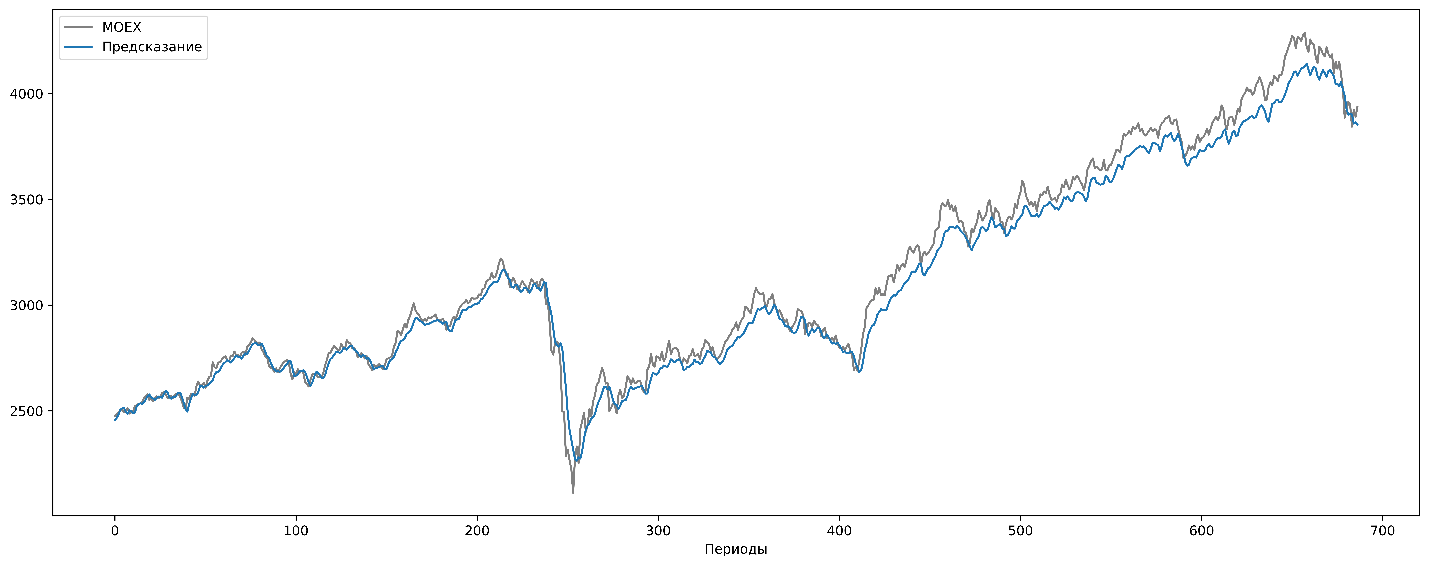


Рисунок 19. Предсказание цены индекса MOEX модели SVR. Источник: расчеты автора.

Модель SVR показывает высокую точность предсказания в начале периода валидации и относительно высокие разрыва к концу периода, что естественно, так как конъектура рынка меняется со временем. Однако, в целом модель хорошо описывает данные.

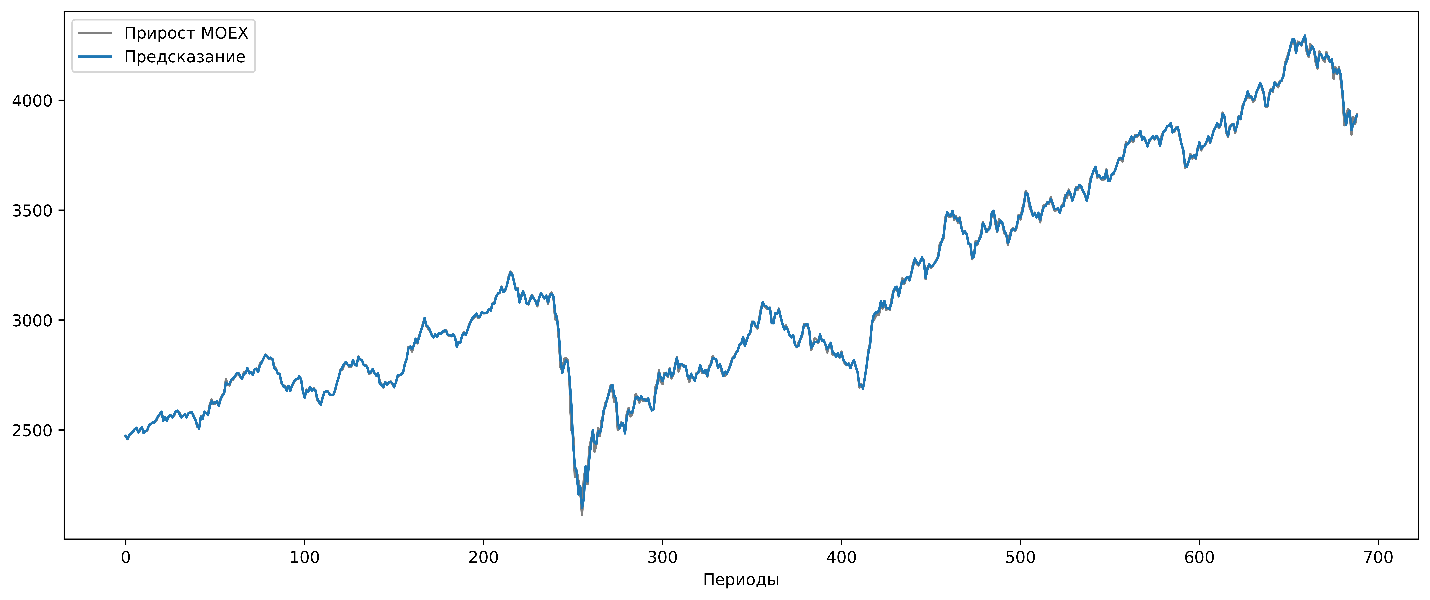


Рисунок 20. Предсказание цены индекса MOEX модели ARIMAX со скользящим окном. Источник: расчеты автора.

Модель ARIMAX со скользящим окном практически идеально описывает данные, иногда недооценивая волатильность. Это естественно, так как модель обучается на самых актуальных данных.

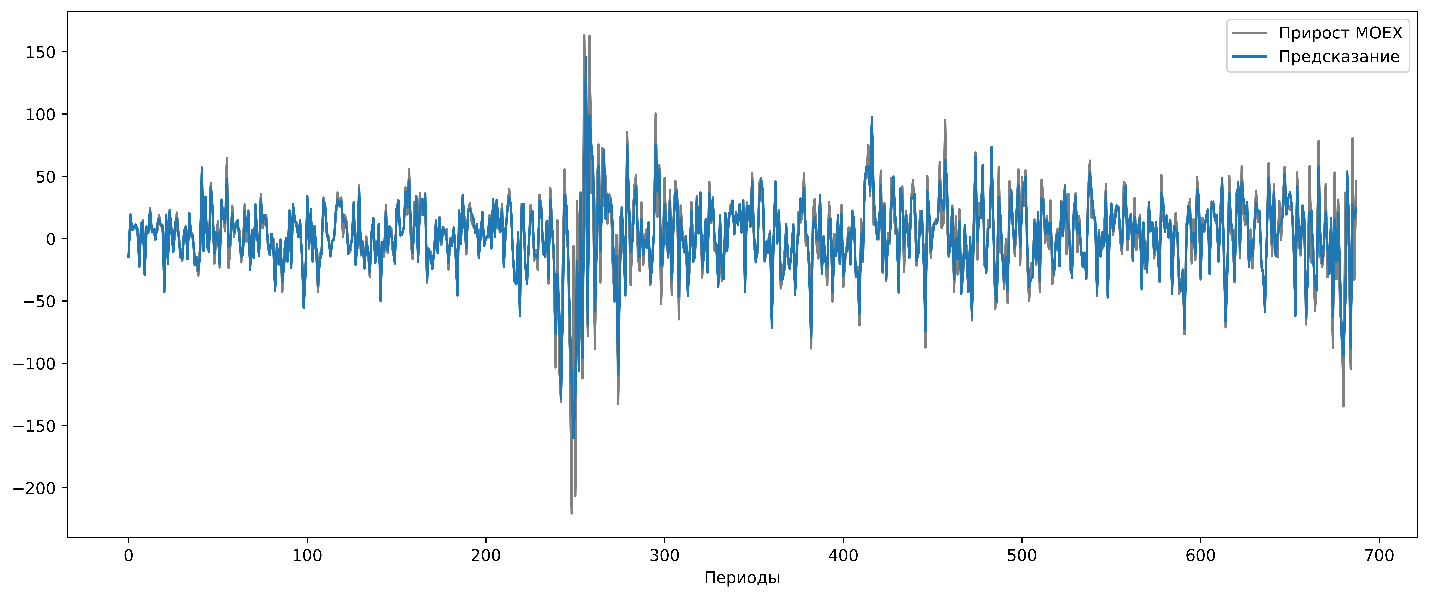


Рисунок 21. Предсказание прироста цены индекса MOEX модели ARIMA-SVR. Источник: расчеты автора.

Данная модель идеально описывает данные особенно в начале валидационного периода. Позднее модель часто недооценивает волатильность прироста цены, однако и сама волатильность возрастает к концу периода. Тем не менее, данная модель визуально лучше всего описывает данные.

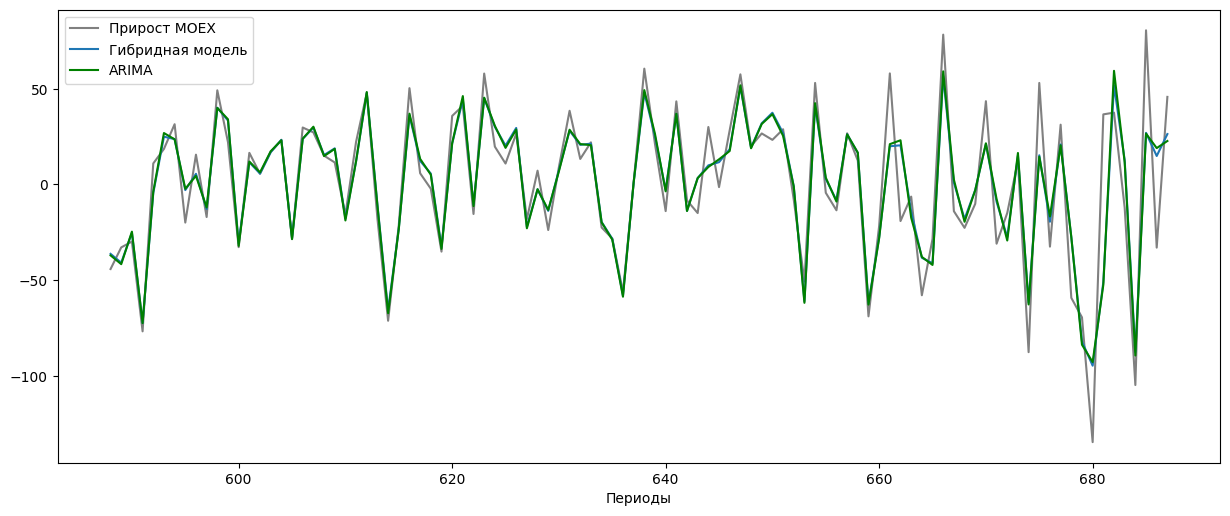
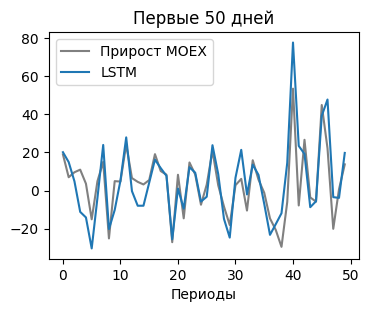


Рисунок 22. Предсказание прироста цены индекса MOEX моделей ARIMA со скользящим окном и ARIMA-SVR в последние 50 дней валидационного периода. Источник: расчеты автора.

Модели хорошо описывают прирост цен даже в последние 50 периодов, но их предсказания практически не отличаются друг от друга. Это говорит о том, что предсказание остатков модели ARIMA с помощью SVR слабо улучшает модель, и в целом не изменяет её.

Изображение выглядит как текст

Автоматически созданное описание

Рисунок 23. Предсказание прироста цены индекса MOEX модели LSTM. Источник: расчеты автора.

На Рисунке 23 представлены предсказания прироста цен в первые и последние 50 дней периода валидации. В то время как в первые недели периода нейронная сеть достаточно хорошо предсказывает прирост цен, в последние дни модель хоть и хорошо описывает динамику приростов, но сильно их переоценивает. Такое поведение модели говорит о том, что скорее всего модель LSTM может быть применима только как модель со скользящим окном.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | MSE | RMSE | R2 | MAPE | MAE |
| ARIMA | 122.51 | 11.07 | **0.9995** | **0.21** | 6.43 |
| SVR | 206.21 | 14.36 | 0.9992 | 0.32 | 9.70 |
| ARIMA-SVR | **116.33** | **10.79** | **0.9995** | **0.21** | **6.29** |
| LSTM | 1823.14 | 42.67 | 0.9927 | 1.03 | 31.91 |
| ARIMA-LSTM | 1200.08 | 34.64 | 0.9953 | 0.80 | 24.74 |
| Наивный прогноз | 687.24 | 26.22 | 0.9987 | 1.1 | 24.74 |

# Результаты

Таблица 8. Метрики качества для прогноза цены индекса MOEX. Источник: расчеты автора.

Как правило, LSTM показывает лучше результаты на развитых рынках чем статистические модели и модели машинного обучения, тем не менее на российском рынке это происходит не всегда. Как показано в Таблице 8, ARIMA со скользящим окном и SVR значительно превышают по качеству модель LSTM. ARIMA со скользящим окном скорее всего имеет преимущество за счет обучения на более актуальных данных чем остальные модели. Разница в качестве моделей LSTM и SVR может быть объяснено наличием большего количества шума на российском рынке чем в развитых странах. LSTM как и все модели глубокого обучения склонна к переобучению и скорее всего слишком подстраивается под случайные колебания цены. Более того, модель глубокого обучения не превышает качество наивного прогноза, что делает её непригодной для использования. В то же время SVR обладает высокой способностью к обобщению, что делает её устойчивой к шуму в данных. ARIMA-SVR показывает наиболее точный прогноз, хотя и не сильно превышающий прогнозы ARIMA. Это показывает, что две данные модели действительно способны дополнять и улучшать друг друга, а также наличие линейных и нелинейных паттернов в динамике цены.

Нужны сопоставления с предыдущими работами

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | MSE | RMSE | R2 | MAPE | MAE |
| ARIMA | **218.31** | **14.77** | 0.7628 | **13.72** | 9.25 |
| SVR | 270.32 | 16.44 | 0.7718 | 20.23 | 11.47 |
| ARIMA-SVR | 251.46 | 15.86 | **0.7878** | 13.89 | **9.06** |
| LSTM | 2118.51 | 46.03 | -0.7795 | 56.71 | 33.23 |
| Наивный прогноз | 1357.58 | 36.85 | -0.9775 | 44.15 | 34.96 |

Таблица 9. Метрики качества для прогноза прироста цены индекса MOEX. Источник: расчеты автора

В таблице 9 показано качество предсказания прироста цен MOEX, с помощью которой можно лучше сравнить предсказательную способность моделей. LSTM остается непригодной для данной задачи. Наивный прогноз также не справляется с данной задачи, так судя по коэффициенту детерминации такой прогноз объясняет дисперсию хуже константы.

Прирост цен прогнозируется хуже, так как на примере наивного прогноза видно, что если модель по каким-либо причинам показывает запаздывающий прогноз, она всё ввиду большего масштаба и

Нужны пояснения, почему ряд цен прогнозируется точно, а прирост цен – нет

Важно отметить, что SVR объясняет дисперсию данных лучше, чем ARIMA со скользящим окном, о чем говорит коэффициент детерминации. Таким образом, модель опорных векторов по одной из метрик способна опережать ARIMA-модель с более актуальными данными.

# Прибыльность торговых стратегий на основе моделей

|  |  |
| --- | --- |
|  | Прибыльность стратегии (%) |
| ARIMA | 522.50 |
| SVR | 509.61 |
| ARIMA-SVR | **522.93** |
| LSTM | 374.90 |
| Наивный прогноз | 51.64 |
| «Buy-and-hold» | 59.63 |

Таблица 9. Прибыльность моделей по стратегии «buy-and-sell». Источник: расчеты автора

Интересны оценки также по двум периодам – обучающему и тестовому

Прибыльности моделей ARIMAX, SVR и ARIMA-SVR составляют более 500% без учета комиссий. Самую высокую прибыльность показывает модель ARIMA-SVR – 522.93%, что является достаточно высоким результатом. Модель LSTM, которая показывала очень низкое качество предсказания прироста цен, всё же может использоваться для создания прибыльной стратегии, а именно с прибыльностью 301%. Это, вероятно, связано с тем, что модель хоть и плохо предсказывает конкретное изменения цены, всё же способна устанавливать направление её движения. Все построенные модели значительно превышают как наивный прогноз, так и стратегию «buy-and-hold», что делает все модели пригодными для использования на фондовом рынке.

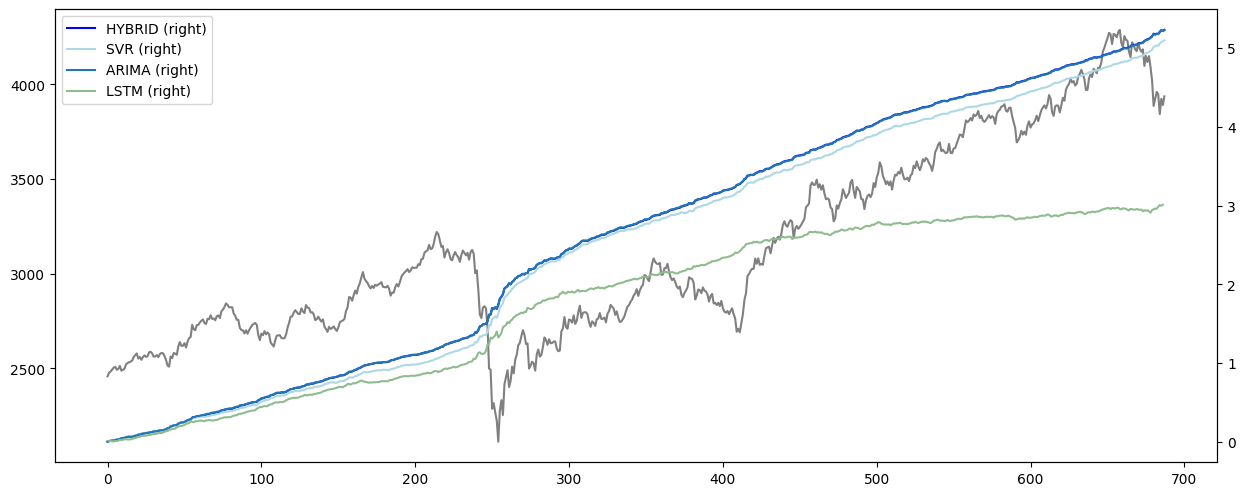


Рисунок 24. Прибыльность моделей по стратегии «buy-and-sell» и график цены акции. Источник: расчеты автора.

На Рисунке 24 видно, что гибридная модель и модель ARIMAX, которые практически совпадают, всё же заметно превышает прибыльность SVR. Однако учитывая то, что модель SVR применяется без использования скользящего окна, это показывает её высокую способность к предсказанию и к обобщению паттернов. Кроме того, в первую половину валидационного периода прибыльность практически совпадает с моделями, использующими скользящее окно. Худшую прибыльность из всех показанных моделей показывает LSTM, однако стратегия всё же выглядит достаточно прибыльной.

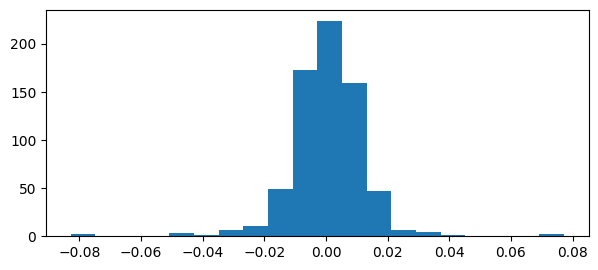


Рисунок 25. Прибыльность сделок модели ARIMA-SVR. Источник: расчеты автора.

Прибыльности сделок модели ARIMA-SVR в основном положительная, но также есть и значительное количество сделок с отрицательной доходностью.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | SVR | HYBRID | ARIMA | LSTM |
| Средняя доходность | 0,7407 | 0,7601 | 0,7594 | 0,4382 |
| Стандартное отклонение доходности | 0,9292 | 0,9129 | 0,9134 | 1,1051 |
| Процент прибыльных сделок (%) | 88,23 | 91,13 | 91,13 | 68,90 |

# Вывод

В данной статье анализируется применение гибридизации моделей ARIMA, SVM и LSTM для прогнозирования фондового рынка. Этому методу уделяется все больше внимания в различных областях, но на российском фондовом рынке и развивающихся рынках в целом не хватает работ, посвященных его изучению. Этот предмет важен, поскольку объединение сложных моделей позволяет значительно улучшить возможности прогнозирования. В целом, изучение нейронных сетей и методов машинного обучения на развивающихся рынках может привести к разработке стратегий с аналогичными показателями прибыльности, которые наблюдаются на развитых рынках.

В целом модель SVR показала отличные результаты, так как она, во-первых, обладает хорошей предсказательной способностью согласно выбранным метрикам. Во-вторых, стратегия, построенная с помощью неё, показывает высокую прибыльность, сопоставимую с моделями скользящего окна. В-третьих, SVR способна улучшить предсказание модели ARIMA со скользящим окном, объясняя её остатки. Наконец, модель опорных векторов показывает больший коэффициент детерминации чем ARIMA со скользящим окном при предсказании прироста цены. Большим плюсом данной модели является её простота в использовании и оптимизации, а также отсутствие необходимости использования метода скользящего окна для получение качественного прогноза.

Модель LSTM в целом способна описывать динамику цены индекса MOEX, а также давать достаточно точные предсказания на короткий период времени. Однако для работы на длительном периоде модель хоть и может предоставить прибыльную стратегию, но предсказательная способность LSTM хуже наивного прогноза. Для применения данной модели глубокого обучения необходимо либо применять метод скользящего окна, либо значительно усложнять модель. Кроме того, возможно создание гибридной модели нескольких нейронных сетей, но подобная модель требует больших усилий, так как нужно не только подобрать комбинацию моделей, но и их параметры. Главный недостаток модели LSTM состоит в сложности оптимизации параметров.

Сопоставляя качество моделей SVR и LSTM, можно сказать о том, что несмотря на большую сложность и продвинутость, модели нейронных сетей всё же склонны к переобучению. В то же время метод опорных векторов благодаря тому, что построен на идее генерализации информации, более устойчив к шуму. Скорее всего, большая зашумленность российского рынка делает SVR более подходящим для российского рынка чем LSTM, хотя на западных рынках модели глубокого обучения, как правило, дают более качественные прогнозы.

# Список литературы

Haykin S (2009) Neural networks and learning machines. Prentice Hall, Upper Saddle River

C. Giovanni, and V. Xavier, "Dynamic Trading and Asset Prices: Keynes vs. Hayek," The Review of Economic Studies, vol. 79, pp. 539–580, November 2011.

Hong H., Jeremy C.S., "A unified theory of underreaction, momentum trading, and overreaction in asset markets," The Journal of finance, vol. 54, pp. 2143–2184, December 1999. G. Soros, The alchemy of finance. John Wiley & Sons, 2015.

Kara Y., Boyacioglu M. Acar, Baykan O.K. Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the istanbul stock exchange Expert Systems with Applications, 38 (5) (2011), pp. 5311-5319

Keynes J.M., The general theory of employment, interest, and money. Macmillan, London, 1936. Neely C.J., Rapach D.E., Jun T., Guofu Z., "Forecasting the Equity Risk Premium: The Role of Technical Indicators." Management Science. vol. 60, pp. 1772–1791, July 2014.

C.J. Neely, D.E. Rapach, T. Jun, and Z. Guofu, "Forecasting the Equity Risk Premium: The Role of Technical Indicators." Management Science. vol. 60, pp. 1772–1791, July 2014.

G. Soros, The alchemy of finance. John Wiley & Sons, 2015.

Treynor J. L., Ferguson R. (1985) ‘In Defense of Technical Analysis’, Journal of Finance (Wiley-Blackwell), 40(3), pp. 757–773. doi: 10.1111/j.1540-6261.1985.tb05000.x.

V. Vapnik, (1995), “The Nature of Statistical Learning Theory”, Springer Verlag.

V. Vapnik, S. Golowich and A. Smola, (1997), “Support Vector Method for Function Approximation, Regression Estimation, and Signal Processing”, in M. Mozer, M. Jordan, and T. Petsche (eds.), Neural Information Processing Systems, Vol. 9. MIT Press, Cambridge, MA.

[1] A.K. Parida, R. Bisoi, P.K. Dash, Chebyshev polynomial functions based locally recurrent neuro-fuzzy information system for prediction of financial and energy market data, J. Finance Data Sci. 2 (3) (2016) 202–223.

[2] Thomas Fischer, Christopher Krauss, Deep learning with long short-term memory networks for financial market predictions, European J. Oper. Res. 270 (2) (2018) 654–669.

[3] Philip Widegren, Deep learning-based forecasting of financial assets, (Master’s thesis), KTH, Mathematical Statistics, 2017.

[4] Anastasia Borovykh, Sander Bohte, Cornelis W. Oosterlee, Dilated convolutional neural networks for time series forecasting, J. Comput. Finance (2018).

[5] Khaled A. Althelaya, El-Sayed M. El-Alfy, Salahadin Mohammed, Evaluation of bidirectional LSTM for short-and long-term stock market prediction, in: 2018 9th International Conference on Information and Communication Systems, ICICS, IEEE, 2018.

[6] Alexiei Dingli, Karl Sant Fournier, Financial time series forecasting–A deep learning approach, Int. J. Mach. Learn. Comput. 7 (5) (2017) 118–122.

[7] Ajit Kumar Rout, P.K. Dash, Rajashree Dash, Ranjeeta Bisoi, Forecasting financial time series using a low complexity recurrent neural network and evolutionary learning approach, J. King Saud Univ. Comput. Inf. Sci. 29 (4) (2017) 536–552.

[8] Gyeeun Jeong, Ha Young Kim, Improving financial trading decisions using deep Q-learning: Predicting the number of shares, action strategies, and transfer learning, Expert Syst. Appl. 117 (2019) 125–138.

[9] Yujin Baek, Ha Young Kim, Modaugnet: A new forecasting framework for stock market index value with an overfitting prevention LSTM module and a prediction LSTM module, Expert Syst. Appl. 113 (2018) 457–480.

[10] Magnus Hansson, On stock return prediction with LSTM networks, 2017.

[11] Li, Z., Han, J. and Song, Y. ‘On the forecasting of high‐frequency financial time series based on ARIMA model improved by deep learning’, Journal of Forecasting, Appl. 39 (2020), pp. 1081–1097.

Anatolyev, S. (2008, January 1). A 10-year retrospective on the determinants of Russian stock returns. RESEARCH IN INTERNATIONAL BUSINESS AND FINANCE, 22(1), 56–67.

Baek, Y. ( 1 ), & Kim, H. Y. ( 1,2 ). (n.d.). ModAugNet: A new forecasting framework for stock market index value with an overfitting prevention LSTM module and a prediction LSTM module. Expert Systems with Applications, 113, 457–480. https://doi-org.proxylibrary.hse.ru/10.1016/j.eswa.2018.07.019

Bao, W. ( 1 ), Rao, Y. ( 1 ), & Yue, J. ( 2 ). (n.d.). A deep learning framework for financial time series using stacked autoencoders and long-short term memory. PLoS ONE, 12(7). https://doi-org.proxylibrary.hse.ru/10.1371/journal.pone.0180944

Dave, E., Leonardo, A., Jeanice, M., & Hanafiah, N. (2021). Forecasting Indonesia Exports using a Hybrid Model ARIMA-LSTM. Procedia Computer Science ; Volume 179, Page 480-487 ; ISSN 1877-0509. https://doi-org.proxylibrary.hse.ru/10.1016/j.procs.2021.01.031

Hansson M. (2017). On stock return prediction with LSTM networks. Master's thesis, Lund.

Kara, Y., Boyacioglu, M. A., & Baykan, Om. K. (2011). Predicting direction of stock price index movement using artificial neural networks and support vector machines: The sample of the Istanbul Stock Exchange. Expert Systems With Applications, 38(5), 5311. https://doi-org.proxylibrary.hse.ru/10.1016/j.eswa.2010.10.027

Khashei, M., & Bijari, M. (2011). A New Hybrid Methodology for Nonlinear Time Series Forecasting. Modelling & Simulation in Engineering, 2011, 1–5. https://doi-org.proxylibrary.hse.ru/10.1155/2011/379121

Li, Z., Han, J., & Song, Y. (2020). On the forecasting of high‐frequency financial time series based on ARIMA model improved by deep learning. Journal of Forecasting, 39(7), 1081–1097. https://doi-org.proxylibrary.hse.ru/10.1002/for.2677

Lozinskaia, A. M., & Zhemchuzhnikov, V. A. (2017). Micex Index Forecasting: The Predictive Power of Neural Network Modeling and Support Vector Machine. Perm University Herald. Economy / Vestnik Permskogo Universiteta. Seria Ekonomika, 12(1), 49–60. https://doi-org.proxylibrary.hse.ru/10.17072/1994-9960-2017-1-49-60

Ma, T., Antoniou, C., & Toledo, T. (2020). Hybrid machine learning algorithm and statistical time series model for network-wide traffic forecast. Transportation Research Part C, 111, 352–372. https://doi-org.proxylibrary.hse.ru/10.1016/j.trc.2019.12.022

Mohan, S., Solanki, A. K., Taluja, H. K., Anuradha, & Singh, A. (2022). Predicting the impact of the third wave of COVID-19 in India using hybrid statistical machine learning models: A time series forecasting and sentiment analysis approach. Computers in Biology and Medicine, 144. https://doi-org.proxylibrary.hse.ru/10.1016/j.compbiomed.2022.105354

Pai, P.-F., & Lin, C.-S. (2005). A hybrid ARIMA and support vector machines model in stock price forecasting. Omega, 33(6), 497.

Perone, G. (2021). Comparison of ARIMA, ETS, NNAR, TBATS and hybrid models to forecast the second wave of COVID-19 hospitalizations in Italy. The European Journal of Health Economics, 1–24. https://doi-org.proxylibrary.hse.ru/10.1007/s10198-021-01347-4

Phan, T.-T.-H., & Nguyen, X. H. (2020). Combining statistical machine learning models with ARIMA for water level forecasting: The case of the Red river. Advances in Water Resources, 142. https://doi-org.proxylibrary.hse.ru/10.1016/j.advwatres.2020.103656

Peresetsky, A. A. (2011). What determines the behavior of the Russian stock market. MPRA Paper.

Petersen, N. C., Rodrigues, F., & Pereira, F. C. (2019). Multi-output Bus Travel Time Prediction with Convolutional LSTM Neural Network. https://doi-org.proxylibrary.hse.ru/10.1016/j.eswa.2018.11.028

Sezer, O. B., Gudelek, M. U., & Ozbayoglu, A. M. (2020). Financial time series forecasting with deep learning: A systematic literature review: 2005–2019. Applied Soft Computing Journal, 90. https://doi-org.proxylibrary.hse.ru/10.1016/j.asoc.2020.106181

Treynor, J. L., & Ferguson, R. (1985). In Defense of Technical Analysis. Journal of Finance (Wiley-Blackwell), 40(3), 757–773. https://doi-org.proxylibrary.hse.ru/10.1111/j.1540-6261.1985.tb05000.x

1. Box, George and Jenkins, Gwilym (1970) Time series analysis: Forecasting and control, San Francisco: Holden-Day. [↑](#footnote-ref-1)