

Министерство образования и науки Российской Федерации
Санкт-Петербургский политехнический университет Петра Великого
Физико-механический институт
Высшая школа теоретической механики и математической физики

Работа допущена к защите

Директор ВШТМиМФ,

д. ф.-м. н., чл.-корр. РАН

А.М. Кривцов

« ___ » _____ 2023 г.

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА
МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ
ПРОГНОЗИРОВАНИЕ ДИНАМИКИ ПАРАМЕТРОВ
РАСПРОСТРАНЕНИЯ НОВЫХ ВИРУСОВ НА ОСНОВЕ
МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ

По направлению 01.04.03 «Механика и математическое моделирование»

Направленность 01.04.03_03 «Механика и цифровое производство»

Выполнил
студент гр.5040103/10301

А.А. Курлевский

Руководитель
доцент ВШТМиМФ, к.ф.-м.н

А.А. Ле-Захаров

Консультант
ассистент ВШТМиМФ

Д.С. Перец

Санкт-Петербург

2023

САНКТ-ПЕТЕРБУРГСКИЙ ПОЛИТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ ПЕТРА ВЕЛИКОГО
Физико-механический институт
Высшая школа теоретической механики и математической физики

УТВЕРЖДАЮ

Директор ВШТМиМФ

А.М. Кривцов

«__» _____ 2023г.

ЗАДАНИЕ

**на выполнение выпускной квалификационной
работы**

студенту Курлевскому Алексею Андреевичу, группы 5040103/10301
фамилия, имя, отчество (при наличии), номер группы

1. Тема работы: прогнозирование динамики параметров распространения новых вирусов на основе методов машинного обучения
2. Срок сдачи студентом законченной работы: 26.05.2023
3. Исходные данные по работе: научные статьи по теме работы, статистика по распространению COVID-19 в России за период с 12.03.2020 до 13.01.2023
4. Содержание работы (перечень подлежащих разработке вопросов): анализ существующих статистических методов для прогнозирования временных рядов, исследование работ по моделированию распространения коронавирусной инфекции, обзор методов машинного обучения в области исследования временных рядов, выбор архитектуры и разработка рекуррентной нейронной сети, оценка результата прогноза
5. Перечень графического материала (с указанием обязательных чертежей): графики с общим количеством заражений, выздоровлений и смертей
6. Консультанты по работе: Д.С. Перец, ассистент ВШТМиМФ

7. Дата выдачи задания 27.02.2023

Руководитель ВКР _____ А.А. Ле-Захаров, доцент ВШТМиМФ, к.ф.-
м.н.
(подпись) инициалы, фамилия

Задание принял к исполнению 27.02.2023
(дата)

Студент _____ А.А. Курлевский
(подпись) инициалы, фамилия

РЕФЕРАТ

На 58 с., 44 рисунка, 5 таблиц

СТАТИСТИЧЕСКИЕ МЕТОДЫ ДЛЯ ПРОГНОЗИРОВАНИЯ ВРЕМЕННЫХ РЯДОВ, ПАНДЕМИЯ COVID-19, LSTM СЕТИ, RNN МОДЕЛИ, ГЛУБОКОЕ ОБУЧЕНИЕ

Тема выпускной квалификационной работы: «Прогнозирование динамики параметров распространения новых вирусов на основе методов машинного обучения»

Целью работы является спрогнозировать динамику характеристики динамического баланса. В ходе работы были рассмотрены наиболее популярные статистические модели такие как ARIMA, SARIMA, метод экспоненциального сглаживания. Для улучшения результата были реализованы модели глубокого обучения. На основе сделанного прогноза проведена оценка количества инфицированных людей, используя балансовую модель CIR.

ABSTRACT

58 pages, 44 figures, 5 tables

STATISTICAL METHODS FOR TIME SERIES FORECASTING, COVID-19 PANDEMIC, LSTM NETWORKS, RNN MODELS, DEEP LEARNING

The topic of the graduation thesis: " Forecasting the progression of patterns of new virus spread using machine learning techniques".

The goal of this research is to forecast the dynamics of the dynamic balance characteristic. Throughout the research, various popular statistical models such as ARIMA, SARIMA, and exponential smoothing methods have been explored and analysed. To enhance the results, deep learning models have been implemented. Based on the prediction made, the number of infected people was estimated using the CIR balance model.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	5
ГЛАВА 1. СТАТИСТИЧЕСКИЕ МОДЕЛИ	8
1.1 ОБЩИЕ КОНЦЕПЦИИ СТАТИСТИЧЕСКИХ МОДЕЛЕЙ.....	8
1.2 ОСНОВНЫЕ ПОНЯТИЯ ВРЕМЕННОГО РЯДА	9
1.3 МЕТОД ХОЛЬТА-ВИНТЕРСА	12
1.4 МОДЕЛИ ARIMA И SARIMA	13
1.5 МОДЕЛИ SIR И SEIR	14
1.6 РЕЗУЛЬТАТ ПРОГНОЗИРОВАНИЯ ХАРАКТЕРИСТИКИ ДБ	16
ГЛАВА 2. МЕТОДЫ МАШИННОГО ОБУЧЕНИЯ	22
2.1 ОБЗОР СУЩЕСТВУЮЩИХ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ	22
2.2 RNN МОДЕЛИ.....	26
2.3 СТРАТЕГИИ ПРОГНОЗИРОВАНИЯ ВРЕМЕННЫХ РЯДОВ.....	30
2.4 ПОДГОТОВКА ДАННЫХ.....	31
2.5 РЕЗУЛЬТАТЫ ПРОГНОЗИРОВАНИЯ RNN МОДЕЛЕЙ.....	34
ГЛАВА 3. DA RNN МОДЕЛЬ	42
3.1 ОБЗОР МОДЕЛЕЙ CODER-DECODER С МЕХАНИЗМОМ ВНИМАНИЯ	42
3.2 ВХОДНОЙ МЕХАНИЗМ ВНИМАНИЯ.....	43
3.3 БЛОК DECODER С ВРЕМЕННЫМ МЕХАНИЗМОМ ВНИМАНИЯ	45
3.4 РЕЗУЛЬТАТЫ ПРОГНОЗИРОВАНИЯ МОДЕЛИ DA RNN.....	46
ГЛАВА 4. ОЦЕНКА КОЛИЧЕСТВА ИНФИЦИРОВАННЫХ ЛЮДЕЙ.	51
4.1 МОДЕЛЬ СВRR.....	51
4.2 БАЛАНСОВАЯ МОДЕЛЬ SIR	56
ЗАКЛЮЧЕНИЕ	62
СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ	64

ВВЕДЕНИЕ

Новая инфекция COVID-19, вызванная коронавирусом SARS-CoV-2, впервые была обнаружена в Китае в декабре 2019 года. Инфекция очень быстро распространилась по всему миру. Уже через 3 месяца, 11 марта 2020 года, всемирная организация здравоохранения объявила ее пандемией. На момент середины апреля 2023 года в мире подтверждено более 682 млн случаев COVID-19, из них 22.7 млн в России

Сложность в принятии решений по мерам профилактики распространения инфекции заключается в отсутствии статистики о распространении вируса за предыдущие периоды времени. Этим же обуславливаются трудности при прогнозировании числа заболевших. Природа вируса не похожа ни на одну из известных ранее заболеваний. Он может оставаться в организме человека около 14 дней [10]. При этом у человека могут не проявляться признаки болезни. Наличие таких особенностей сильно усложняет анализ имеющихся данных о параметрах распространения инфекции. Ученые всех стран разрабатывают механизмы краткосрочного и долгосрочного прогнозирования динамики заболеваемости COVID-19. Например, исследователям [4] удалось успешно применить модель SEIR для моделирования распространения вируса в Индонезии. Результаты модели точно описывают число активных случаев и также показывают, что вакцинация ускоряет процесс выздоровления, а меры изоляции замедляют распространение вируса. В статье [16] авторы строили прогноз для многих стран, используя модель ARIMA. Для европейских стран средняя квадратичная ошибка числа подтвержденных случаев составила $2.88 \cdot 10^4$ при прогнозировании динамики на 2 недели.

Однако, классические статистические методы прогнозирования довольно ограничены числом обучаемых параметров. Из-за этого вероятность того, что модель обучилась не до конца возрастает. Это будет заметно, когда количество наблюдений резко возрастет. Также, при увеличении горизонта предсказания, качество модели ухудшается. На смену традиционным моделям

приходят нейронные сети и другие модели машинного обучения. Еще недавно специалисты считали их неконкурентноспособными. В основном критика была вызвана из-за невозможности интерпретации результата прогноза. В нейронных сетях количество обучаемых параметров может достигать более 1 млн. Такое свойство позволяет модели быть более гибкой и делать качественные прогнозы. Но с другой стороны процесс обучения модели сильно усложняется и требует много вычислительных ресурсов. Оценить как модель себя поведет на тех или иных данных становится невозможным. Тем не менее, модели машинного обучения пользуются большой популярностью у исследователей и показывают неплохой результат. Используя рекуррентные нейронные сети, авторам [13] удалось спрогнозировать число подтвержденных случаев с точностью 87%. В [5] средняя абсолютная ошибка краткосрочного прогноза (1-3 дня) того же самого показателя составила менее 3%, а у недельного прогноза ошибка лежит в пределах 4%-8%.

В 2021 году была предложена новая модель распространения эпидемии SIR [1], учитывающая показатели пандемии в текущий и прошлые моменты времени. Модель предсказала траекторию числа активных больных с отклонением 1.13% на интервале с 06.06.2020 по 30.06.2020. В работе [1] впервые сформулирован принцип динамического баланса эпидемиологического процесса. Идея этого принципа основывается на гипотезе близости значений общего количества заболевших в прошлом и значений общего количества выздоровевших и умерших в текущий момент времени. Визуальное представление принципа динамического баланса представлено на рисунке 1.

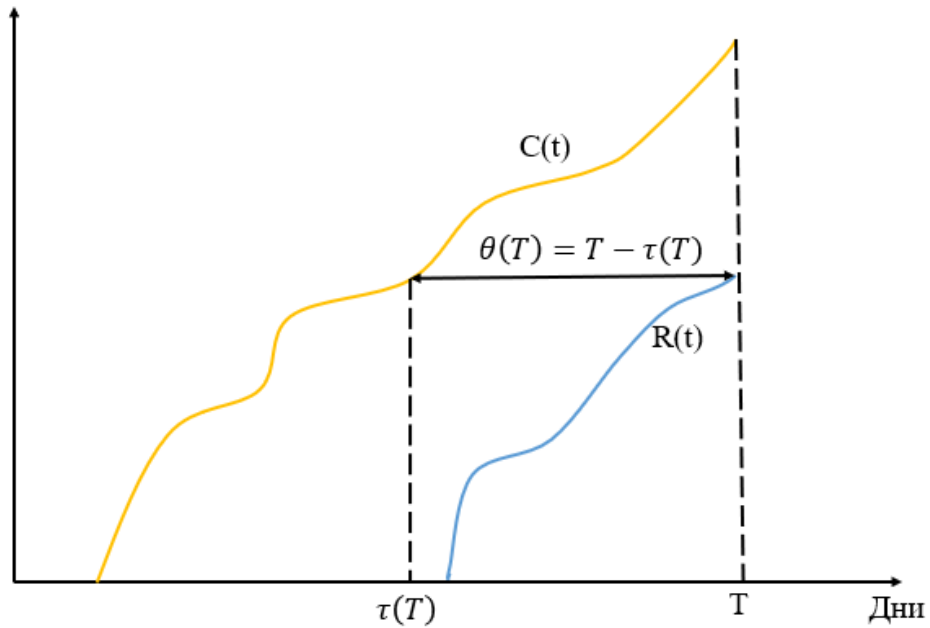


Рисунок 1. Принцип динамического баланса.

Где $C(t)$ – общее количество заболевших, $R(t)$ – количество выздоровевших и умерших, $\theta(T)$ – характеристика динамического баланса (ХДБ)

Цель данной работы – научиться предсказывать ХДБ методами машинного обучения. Это позволит сделать прогноз более точным и увеличит горизонт прогнозирования модели. В ходе работы будут решены следующие задачи:

1. Провести анализ существующих моделей прогнозирования. Подчеркнуть основные идеи, заложенные в построении алгоритма предсказания
2. Предложить метод машинного обучения для нахождения зависимости между значениями параметров распространения и характеристики динамического баланса (ХДБ).
3. На основе сделанного прогноза оценить количество болеющих людей, используя принцип динамического баланса.

ГЛАВА 1. СТАТИСТИЧЕСКИЕ МОДЕЛИ

1.1 ОБЩИЕ КОНЦЕПЦИИ СТАТИСТИЧЕСКИХ МОДЕЛЕЙ

Развитие вычислительных технологий, включая методы обработки и хранения больших объемов данных, предоставили массу средств для изучения. Информация может быть представлена в самых разных форматах. Начиная от базовых, таких как численные и текстовые, до более сложных аудио и видео. Методы хранения, управления и сбора данных, изменяющихся со временем, вызывают отдельный интерес у исследователей и предоставляют инженеру очень сложную задачу по созданию и проектированию баз данных. Все эти достижения и исследования порождают такую большую отрасль как интеллектуальный анализ временных рядов. Предсказание динамики изменения данных – это одна из задач, решаемая в данной отрасли. Мотивация прогнозирования очень проста – сократить неопределенность в будущем, помочь принять решение на основе спрогнозированных данных. Важность прогноза особенно выделяется у динамических явлений: цена закрытия акций компании, уровень безработицы, энергетическая нагрузка, прогноз погоды и многие другие области.

Методы прогнозирования временных рядов основаны на идее, что данные имеют исторические закономерности, которые можно использовать для описания будущего. Очень важно, чтобы последовательность значений была измерена через равные промежутки времени. Как правило эти закономерности нетривиальны и их обнаружение является основной целью в обработке временных рядов. При изучении данных необходимо найти такие условия, при которых найденная зависимость повторяется и при каких изменениях сделанное предположение не будет верно.

Структура модели прогнозирования проектируется таким образом, чтобы алгоритм максимально подробно описывал процесс. Модель запоминает задействованные переменные, основываясь на сделанных предположениях, и формулирует динамическую связь между ними, собирая

их в одну большую и потенциально математически гибкую структуру. Сформулированные предположения можно использовать как для предсказания, так и для исследования природы данных.

Статистические методы, основанные на авторегрессии и скользящих средних, остаются современными методами для моделирования временных рядов уже на протяжении полувека. Алгоритмы, используемые в таких методах, предполагают, что данные распределены в соответствии с уже известным распределением. Использование таких методов требует сложного математического аппарата и обширные технические знания для подбора параметров. Поэтому, прежде чем давать описание моделям рассмотрим из чего состоит временной ряд.

1.2 ОСНОВНЫЕ ПОНЯТИЯ ВРЕМЕННОГО РЯДА

Временной ряд X размера n – это упорядоченная последовательность наблюдений, т. е. $X = (x_1, x_2, x_3, \dots, x_n)$, где x_t – наблюдение в момент времени t , измеренных через равный промежуток времени.

Если значения ряда описываются функцией $y = f(t)$, то такой ряд называется детерминированным. В нем отсутствуют случайные факторы. Если же помимо математической функции ряд содержит случайный член ε , $y = f(t, \varepsilon)$, то такой ряд называется стохастическим или недетерминированным.

Другое важное свойство – стационарность. Это свойство, при котором, не меняются основные характеристики ряда (дисперсия и математическое ожидание) с течением времени. Эталонным стационарным рядом считается белый шум. Ниже приведено графическое представление стационарности.

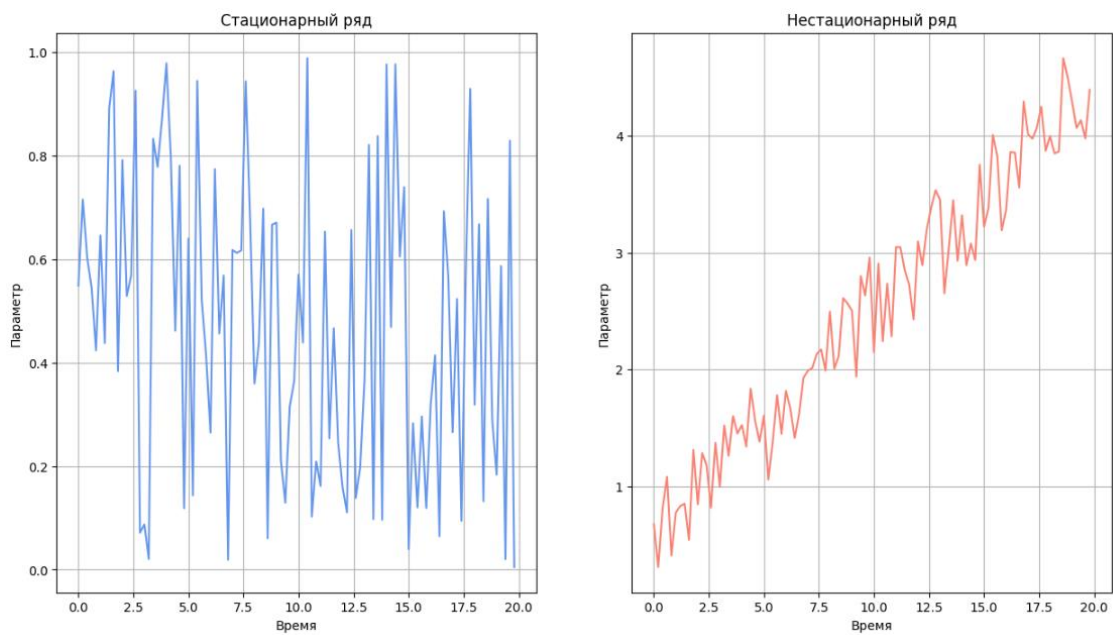


Рисунок 2. Сравнение стационарного и нестационарного ряда.

На рис.2 слева представлен стационарный ряд. Справа у ряда изменяется его математическое ожидание. Поэтому ряд не является стационарным.

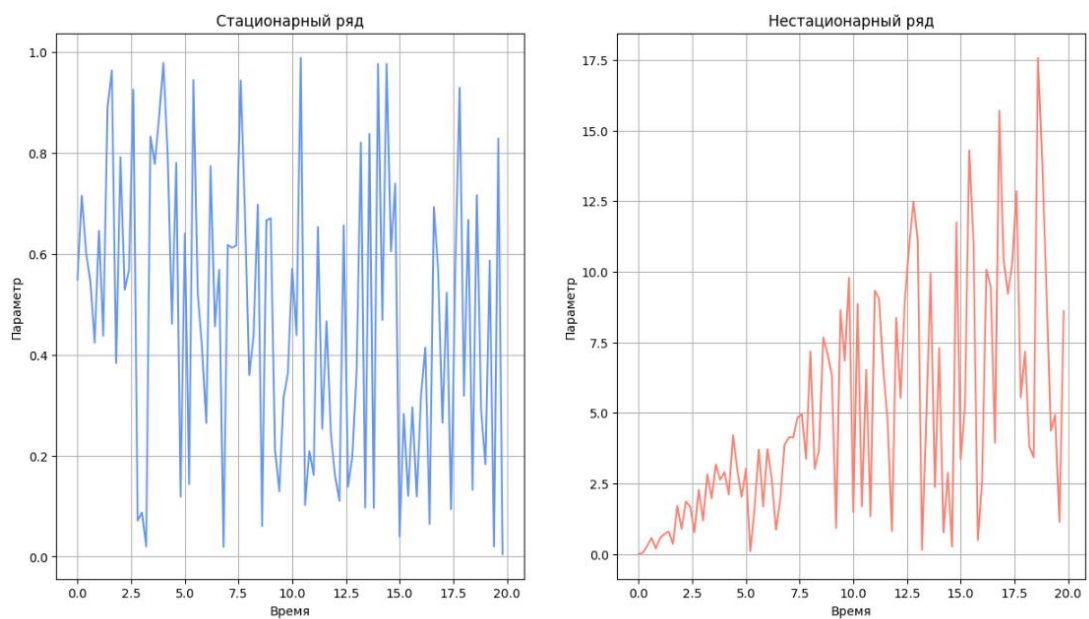


Рисунок 3. Сравнение стационарного и нестационарного ряда. Нарушение гомоскедастичности

На рис. 3 также приведено сравнение двух последовательностей. Однако помимо того, что у ряда справа непостоянно математического ожидание,

видно, что разброс значений разный в зависимости от времени. Нарушено свойство постоянства дисперсии (гомоскедастичность).

Для большинства статистических моделей наличие стационарности является ключевым фактором, так как они используют основные свойства ряда (математическое ожидание, дисперсия). Для проверки ряда на стационарность используется тест Дики-Фуллера [24]. Если же ряд не является стационарным, то вычисляют первую разность $\Delta x_t = x_t - x_{t-1}$. Как правило, данного преобразования достаточно, чтобы привести ряд к стационарному виду.

На практике очень редко встречаются временные последовательности, которые обладают стационарностью. У реальных процессов может наблюдаться возрастающее или убывающее поведение, сезонный период, в котором характер динамики сохраняется на протяжении некоторого времени. Для изучения этих свойств ряд декомпозируют на несколько компонент. Это позволяет не только получить более подробную информацию о данных, но и извлечь важные свойства для вычислительной модели. Принято выделять 3 основные компоненты ряда.

1. Тренд (T). Это долговременная систематическая компонента, которая описывает общую тенденцию. Тренд может иметь самый разнообразный характер (линейный, экспоненциальный, полиномиальный и др.)
2. Сезонность (S). Это циклические колебания, которые повторяются через относительно постоянный временной интервал. Примером сезонности является увеличение электропотребления в зимнее время, увеличение продаж в выходные и праздничные дни.
3. Остаток (R). Это та часть ряда, которая не поддается систематическому описанию и является непредсказуемой. Обычно непредсказуемость вносят такие факторы как политическая обстановка, стихийные бедствия, кризисы и др.

Теперь, когда введены основные компоненты, можно сформулировать 2 основных представления временного ряда:

$$X_t = T_t + S_t + R_t$$

Описание выше соответствует аддитивному представлению ряда. В нем исследуемая величина представлена в виде суммы всех компонент. Другое представление – мультипликативное:

$$X_t = T_t \times S_t \times R_t$$

При таком описании ряд представляет собой произведение всех трех компонент. Наибольший вклад вносит трендовая составляющая, а остальные компоненты изменяют его. Важно заметить, что не все явления описываются согласно приведенным формулам.

Далее рассмотрим несколько наиболее популярных статистических моделей, применяемых для прогнозирования.

1.3 МЕТОД ХОЛЬТА-ВИНТЕРСА

Метод сезонного экспоненциального сглаживания Хольта-Винтерса состоит из трех уравнений с параметрами сглаживания, связанных с основными компонентами временного ряда. Учитывая такую схему, метод может использоваться в двух видах: мультипликативный и аддитивный.

Авторы [23] утверждают, что метод Хольта-Винтерса может быть применен для регулировки временных рядов, в которых амплитуда сезонных колебаний возрастает с увеличением среднего значения. Мультипликативная модель описывается следующими выражениями:

$$L_t = \alpha \frac{z_t}{S_{t-s}} + (1 - \alpha)(L_{t-1} + T_{t-1})$$

$$T_t = \beta(L_t - L_{t-1}) + (1 - \beta)T_{t-1}$$

$$S_t = \gamma \frac{z_t}{L_t} + (1 - \gamma)S_{t-s}$$

$$z_{t+h} = (L_t + hT_t)S_{t-s+h}$$

Где α, β, γ – параметры сглаживания; s – определяет количество наблюдений, входящих в сезонный период; z_{t+h} – прогноз значение для периода $t + h$.

В [12] авторы строили прогноз для общего количества выявленных случаев COVID-19 в Горонтало, провинции Индонезии. Средняя абсолютная ошибка в процентах изменялась в пределах от 6.14% до 6.56% при горизонте прогнозирования с 10 апреля по 13 октября 2020 года. В работе [21] было проведено сравнение метода Хольта-Винтерса с моделью ARIMA. Исследования показали, что для ковида результат предсказания лучше у модели ARIMA.

1.4 МОДЕЛИ ARIMA И SARIMA

Модель ARIMA(p, d, q) была предложена Боксом и Дженкинсом в 1970-х годах [7] для анализа временных рядов. В ее аббревиатуре скрыты 3 ключевых операции. AR(p) – авторегрессия часть. Показывает сколько значений в прошлом используется для прогнозирования будущего значения. I(d) – интеграционная часть. Показывает на сколько отличаются необработанные наблюдения. Данная операция позволяет привести ряд к стационарному виду, d – степень разности. MA(q) – скользящее среднее. Эта часть добавляет зависимость между наблюдением и остаточной ошибкой из модели скользящего среднего, q – порядок скользящего среднего.

Модель ARIMA для нестационарного временного ряда X_t имеет вид:

$$\Delta^d X_t = c + \sum_{i=1}^p \alpha_i \Delta^d X_{t-i} + \sum_{j=1}^q b_j \varepsilon_{t-j} + \varepsilon_t$$

Где ε_t – стационарный временной ряд; X_t – нестационарный ряд; α_i и b_j параметры, которые отвечают соответственно за: авторегрессию с длиной лага p и скользящее среднее с длиной лага q ; Δ^d – оператор разности временного ряда порядка d ; c – описывает начальный уровень модели (аналог свободного члена в линейной регрессии)

ARIMA применяется для моделирования однородных нестационарных процессов. Большинство исследуемых данных обладают сезонностью. Поэтому для улучшения результата прогноза логично использовать автокорреляцию не только для последовательных моментов, но и внутри сезона. Такой механизм используется в модификации сезонной модели ARIMA, также известной как SARIMA. По своей структуре модели очень похожи, только помимо несезонной части, описанной выше, в SARIMA есть часть, которая учитывает сезонную компоненту ряда. Формально эта часть описывается следующим уравнением:

$$\Delta^D X_t = c + \sum_{i=1}^P A_{is} \Delta^D X_{t-is} + \sum_{j=1}^Q B_{js} \varepsilon_{t-js} + \varepsilon_t$$

Здесь все параметры аналогичны параметрам для несезонной части. Единственная разница – это наличие порядка сезонности s . Результат модели SARIMA это сумма сезонной и несезонной части.

По своему описанию модели подходят для прогнозирования динамики распространения COVID-19. К примеру, в работе [27] авторы сравнивали ARIMA с моделью Prophet. Средняя абсолютная ошибка для предсказания общего количества случаев в промежутке с 01.05.2020 по 30.11.2021 в США составляет 1847.39, а средняя относительная ошибка 0.00003, что ниже, чем у Prophet. Для Бразилии средняя абсолютная ошибка составила 94.22 (ARIMA) и 2370.45 (Prophet). Но при моделировании количества заболевших за день прогноз SARIMA хуже прогноза Prophet. Авторы утверждают, что для исследования ежедневных случаев, динамики с сезонностью и большим разбросом в данных, больше подходит модель Prophet. В рамках данной работы будут показаны результаты ARIMA и SARIMA.

1.5 МОДЕЛИ SIR И SEIR

Модели SIR и SEIR по своей концепции относятся к моделям камерного типа. Такой метод моделирования распространения эпидемии рассматривает отдельную популяцию $N(t)$, которую разделяют на 3 «камеры»: $S(t)$ – класс

восприимчивых, $I(t)$ – класс зараженных, $R(t)$ – класс выздоровевших. Каждая группа представляется в виде дифференциальных уравнений:

$$\frac{dS(t)}{dt} = -\frac{\alpha S(t)I(t)}{N(t)}$$

$$\frac{dI(t)}{dt} = \frac{\alpha S(t)I(t)}{N(t)} - \beta I(t)$$

$$\frac{dR(t)}{dt} = \beta I(t)$$

$$N(t) = S(t) + I(t) + R(t)$$

Где α и β – параметры, показывающий скорость перехода из группы восприимчивых в инфицированную и из группы инфицированных в выздоровевшую соответственно.

SIR считается мощным и гибким инструментом для анализа болезней. В работе [3] авторы исследовали распространение COVID-19 в городах США за период с 10 марта 2020 по 7 марта 2021 года. Метрика R^2 достигала значений 0.9873 при прогнозировании группы выздоровевших. Для других классов порядок метрики такой же. Это говорит о том, что модель очень хорошо предсказывает тренд динамики.

Другой популярной камерной моделью является SEIR. Она похожа на SIR. Отличие состоит в том, что добавляется класс E – группа людей, которые были инфицированными, но еще не являются заразными. Модель описывается системой ДУ:

$$\frac{dS(t)}{dt} = -\frac{\beta S(t)I(t)}{N(t)} + \xi R(t)$$

$$\frac{dE(t)}{dt} = -\frac{\beta S(t)I(t)}{N(t)} - \sigma E(t)$$

$$\frac{dI(t)}{dt} = \sigma E(t) - \gamma I(t)$$

$$\frac{dR(t)}{dt} = \gamma I(t) - \xi R(t)$$

Где β – вероятность заболевания, которое передастся восприимчивому человеку от инфицированного, σ – скорость, при которой человек становится инфекционным, γ – скорость выздоровления, ξ – скорость, с которой выздоровевшие люди становятся восприимчивы в следствии низкого иммунитета или других проблем со здоровьем.

В исследовании [22] авторы сравнивали SEIR с регрессионной моделью на примере распространения COVID-19 в Индии. Среднеквадратичная логарифмическая ошибка прогнозирования общего числа подтверждённых случаев за 2 недели составила 1.75 и 1.52 для регрессионной модели и SEIR соответственно.

1.6 РЕЗУЛЬТАТ ПРОГНОЗИРОВАНИЯ ХАРАКТЕРИСТИКИ ДБ

Перейдем непосредственно к исследованию временного ряда статистическим методами. Динамика прогнозируемой величины (ХДБ) представлена на рис.4

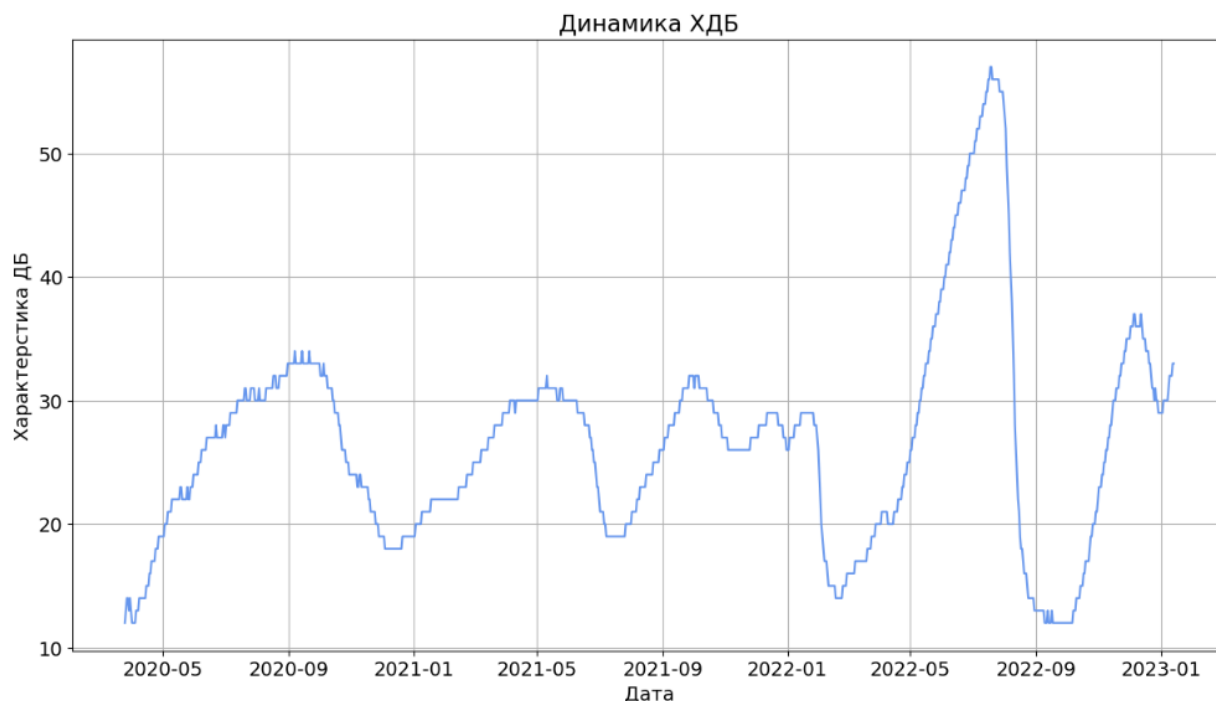


Рисунок 4. Динамика ХДБ в России.

По графику на рис. 4 можно сделать вывод о том, что ряд нестационарный, прослеживается сезонность, в промежутке с сентября по май 2022 года присутствует явный выброс.

Как говорилось ранее, прежде чем применять модели необходимо провести анализ временного ряда. Для начала найдем степень полинома, который наилучшим образом описывает представленную динамику. Оценивать результат будем по 4 метрикам: RMSE, MAE, R^2 , MAPE, вычисляемых по формулам:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n}}$$

$$MAE = \frac{\sum_{i=1}^n |y_i - \hat{y}_i|}{n}$$

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

$$MAPE = \frac{1}{n} \sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{y_i}$$

Где y_i – фактическое значение, \hat{y}_i – результат предсказания, $\bar{y} = \frac{1}{n} \sum_{i=1}^n y_i$ – среднее значение динамики, n – количество наблюдений.

Результаты подбора степени полинома:

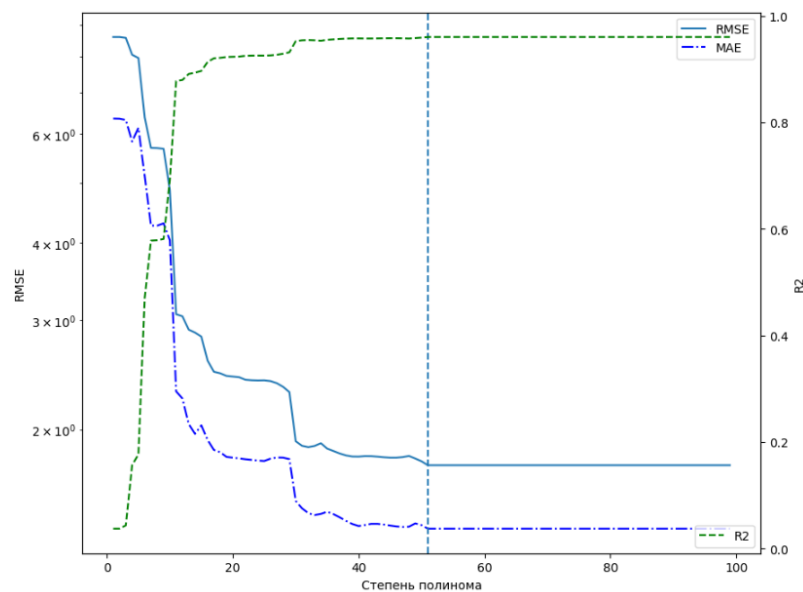


Рисунок 5. Выбор лучшей степени полинома

Метрика R^2 показывает, насколько хорошо тренд прогнозных значений совпадает с истинным трендом. Если значение равно 1, то прогноз можно считать идеальным. Если 0, то тренд динамики предсказывается очень плохо. Другие метрики иллюстрируют количественную точность. Чем они меньше, тем прогноз лучше. Поэтому логика выбора степени полинома такая: выбирается самое оптимальное значение, при котором R^2 наибольшее, а MAE и $RMSE$ наименьшие. Тогда по рис.5 можно сделать вывод, что полином 51 степени наилучшим образом описывает динамику. Уберем полиномиальный тренд и рассмотрим остатки ряда.

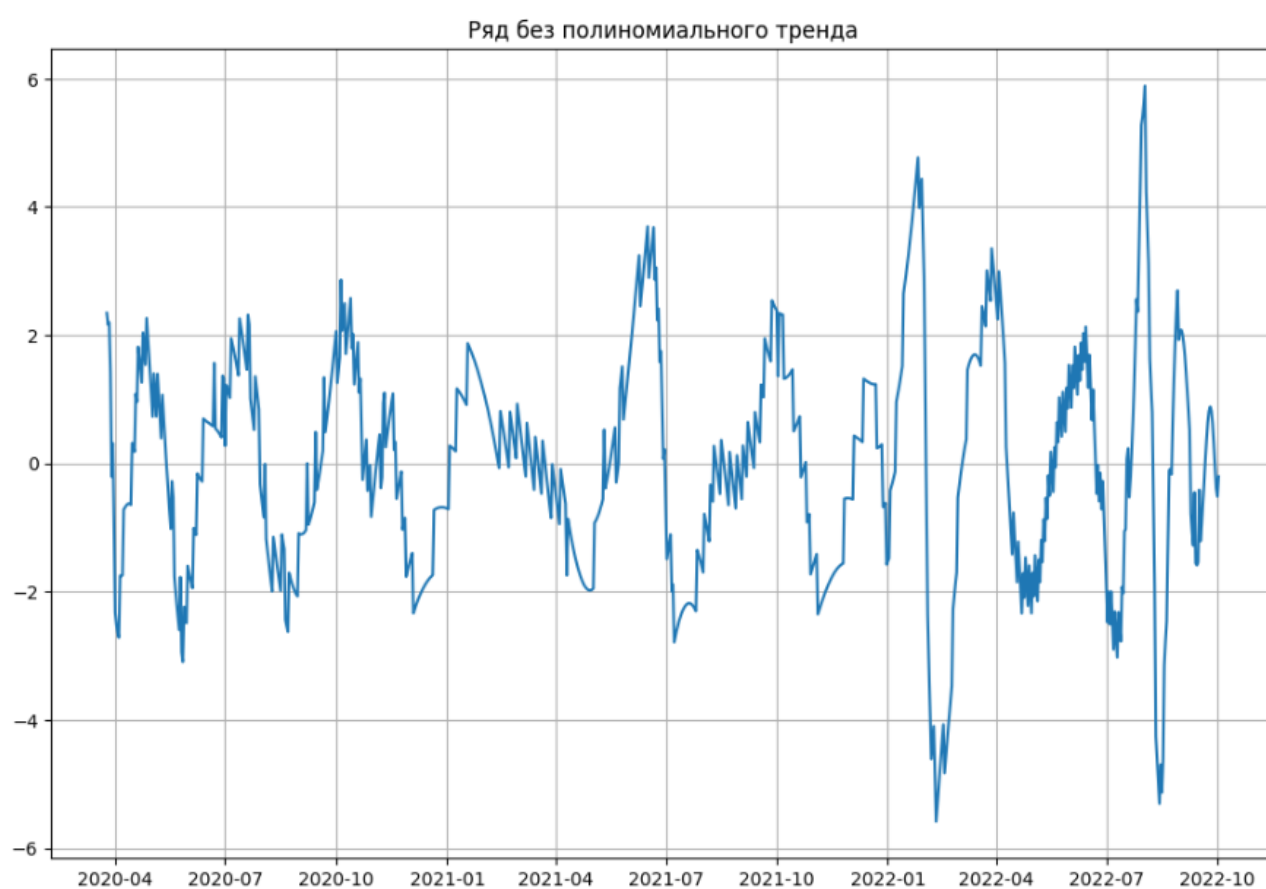


Рисунок 6. Ряд без полиномиального тренда

На графике более отчетливо прослеживается сезонная составляющая ряда. На мой взгляд, период сезона равен 90 дней. Из-за наличия сезона нельзя сделать вывод о том, что ряд стационарен, хотя критерий Дики-Фуллера говорит об обратном. Возможно, для критерия требуется больше наблюдений,

чтобы сделать достоверный вывод. Вычислим первую разность ряда и посмотрим, как выглядит график частичной и полной автокорреляции:

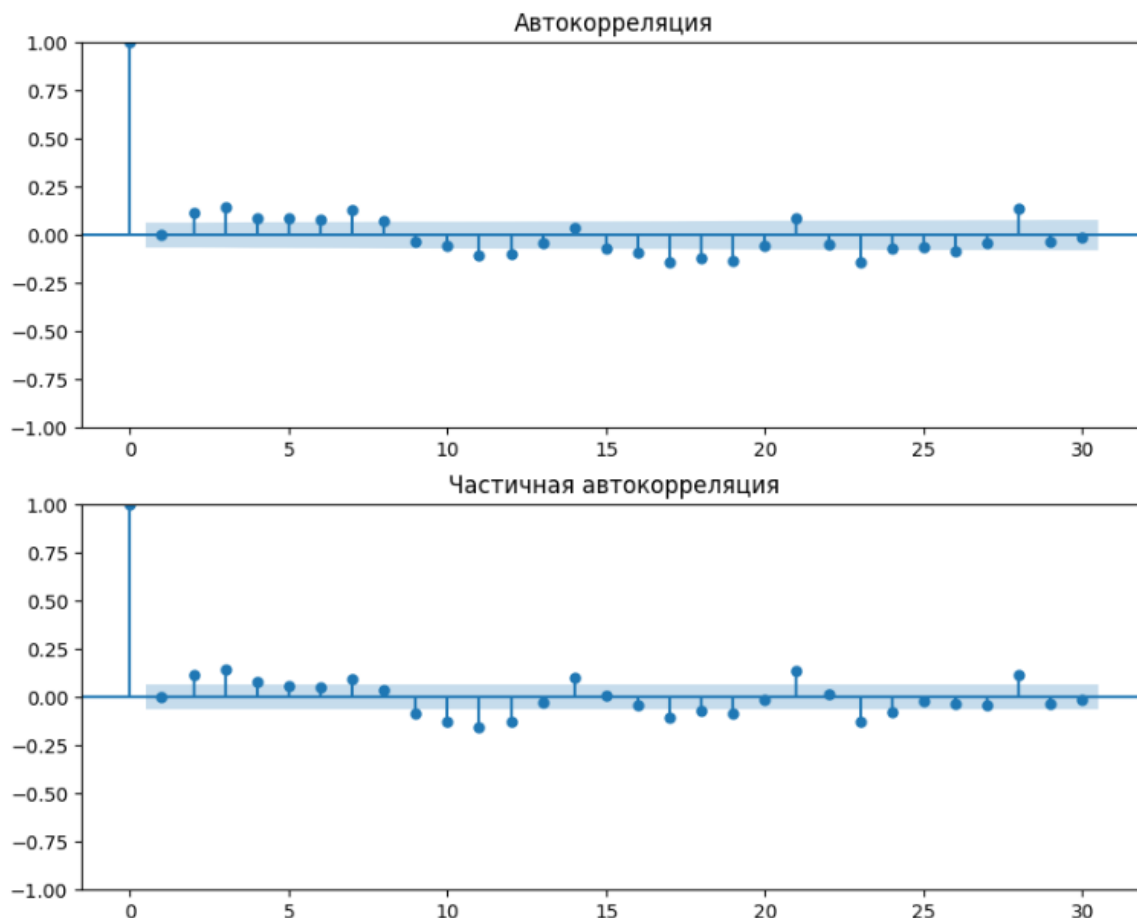


Рисунок 7. Графики частичной и полной автокорреляции

По графикам, изображенным на рис.7, невозможно однозначно определить параметры для моделей ARIMA и SARIMA. Однако, первоначальное приближение все-таки стоит сделать. По моему мнению, оптимальные параметры будут следующие: $p=10$, $d=1$, $q=9$, $P=11$, $D=1$, $Q=8$, $s=90$. Подбор параметров для моделей осуществлялся перебором по сетке (GridSearch). Результаты предсказания представлены ниже:

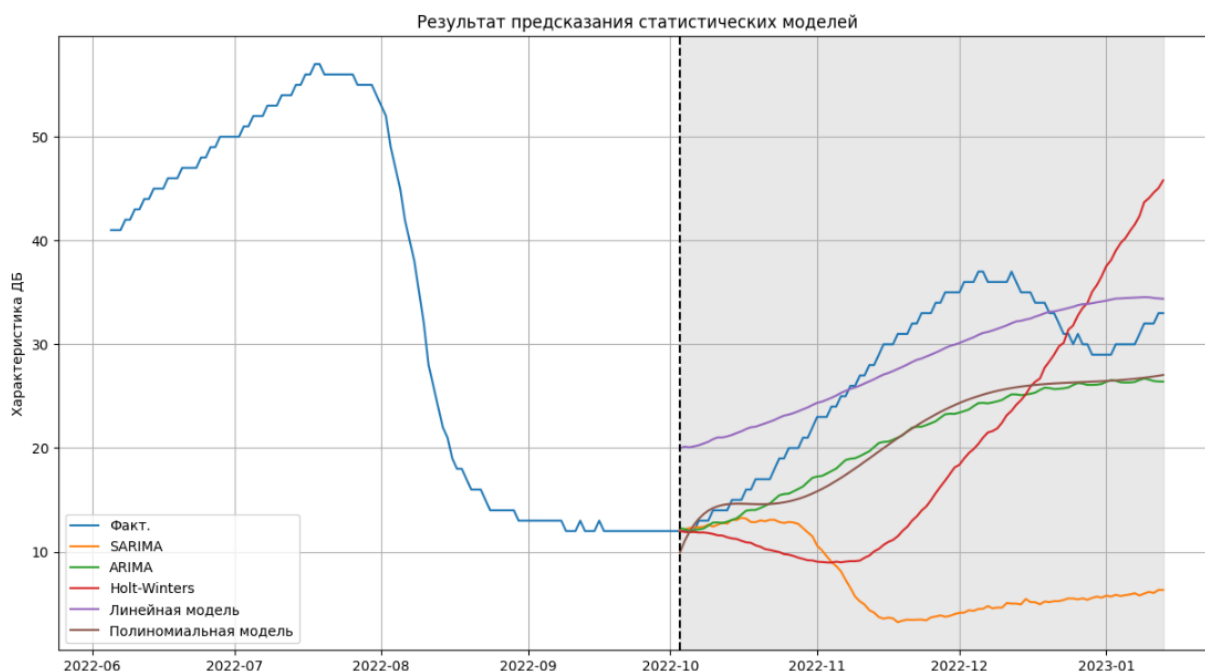


Рисунок 8. Предсказание статистических моделей в промежутке с 03.10.2022 по 13.01.2023 (103 дня).

Модель	MAPE, %	R ²
Линейная	18.03	0.67
Полиномиальная	21.15	0.13
ARIMA	21.27	0.11
Holter-Winters	38.35	-1.55
SARIMA	64.76	-7.64

Таблица 1. Метрики качества предсказания для статистических моделей

Результат получился довольно неожиданным. Лучшей оказалась модель, которая требует меньше всего временных ресурсов и использует наиболее простой математический аппарат – линейная регрессионная модель. Далее очень похожие результаты показали полиномиальная модель и ARIMA. Первая модель лучше предсказывает величину прогнозируемого параметра, а вторая тренд. Далее идет метод экспоненциального сглаживания. На мой взгляд, у модели довольно неплохое значение MAPE. Но коэффициент детерминации говорит, что тренд предсказания ХДБ далек от фактического. Худшей оказалась модель SARIMA. Но нужно учесть тот фактор, что для этой модели не удалось рассмотреть большое количество комбинаций параметров. Их валидация требует большие вычислительные ресурсы и занимает довольно

значительное время для обучения. Возможно, если уделить модели больше внимания, то результат улучшится.

Подводя итог данной главы, можно сделать вывод, что с помощью статистических моделей можно добиться качественного и точного прогноза как на больших, так и на малых горизонтах прогнозирования. Это подтверждают и исследования, упомянутые в главе, и собственные результаты. Помимо этого, для большинства моделей не требуется много времени на обучение. Но у статистических моделей присутствуют значительные недостатки:

1. Не всегда удастся учесть дополнительные внешние факторы. Во многих случаях удастся найти корреляцию между исследуемым параметром и другим процессом. Учет данной зависимости позволит улучшить прогноз
2. При долгосрочном прогнозе качество модели значительно ухудшается. Обычно прогноз устремляется к среднему значению. Поэтому использовать такие модели на большом промежутке времени невозможно. Требуется заново проводить исследования.

Модели машинного обучения решают приведенные недостатки. Однако, в таких моделях заложен еще более сложный математический аппарат и требуется больше времени для оптимального подбора гиперпараметров. Исследованию большой отрасли машинного обучения (рекуррентные нейронные сети) посвящена оставшаяся часть работы.

ГЛАВА 2. МЕТОДЫ МАШИННОГО ОБУЧЕНИЯ

2.1 ОБЗОР СУЩЕСТВУЮЩИХ МЕТОДОВ МАШИННОГО ОБУЧЕНИЯ

За последние 2 десятилетия модели машинного обучения привлекли к себе огромное количество внимания и зарекомендовали себя как серьёзные соперники классических статистических моделей в сообществе прогнозистов. Из-за своей структуры модели часто называют моделями черного ящика или моделями, управляемыми данными. Причин для таких названий несколько. Алгоритмы машинного обучения учитывают огромное количество параметров при предсказании. Дать логическое объяснение почему получился тот или иной результат невозможно. Как правило, исследователи используют эмпирические знания при выборе алгоритма.

Модели машинного обучения относятся к классу непараметрических нелинейных моделей. Для построения прогноза не требуется предварительного знания о распределении исследуемой величины. Алгоритмы явно не зависят от параметров. Обучение происходит на основе исторических данных, тем самым выявляются зависимости между прошлыми и будущими значениями. Благодаря этому свойству модели способны адаптироваться под широкий класс задач и находить нелинейный характер в исследуемых данных.

Описанные преимущества предоставляют огромный потенциал в использовании методов машинного обучения для прогнозирования временных рядов. В [28] утверждается, что искусственные нейронные сети превосходят классические статистические методы такие как линейная регрессия и ARIMA модели. На соревнованиях по прогнозированию (например, NN3 [11]) модели машинного обучения показывают лучший результат при различных условиях.

Искусственные нейронные сети (ИНС) являются одними из самых популярных методов машинного обучения. При создании ИНС авторы

вдохновлялись обработкой информации человеческим мозгом. Простейшей формой нейронной сети является перцептрон (аналог нейрона). Схема перцептрона представлена на рис. 9

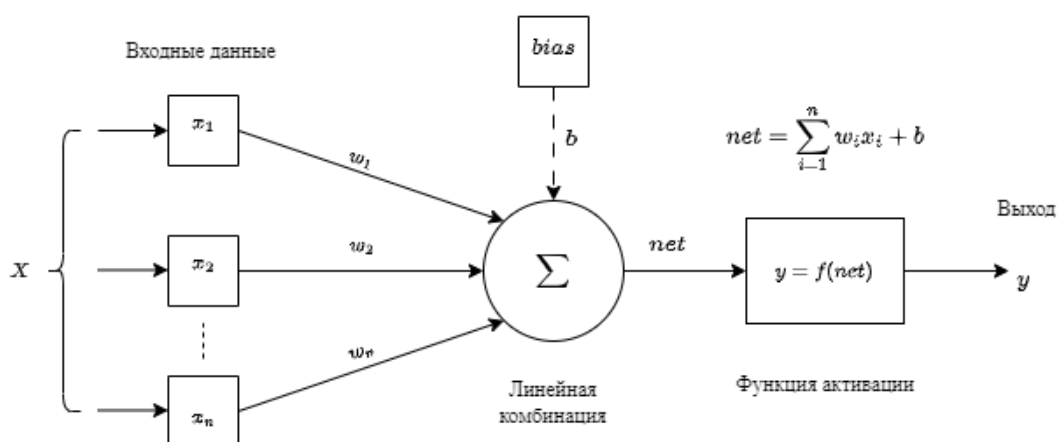


Рисунок 9. Схема перцептрона

На вход перцептрону подается вектор входных значений X . Каждой компоненте вектора x_i соответствует весовой коэффициент w_i . Результат скалярного произведения вектора X и вектора W с добавлением bias (аналог интерцептора в линейной регрессии) передается в функцию активации, которая вычисляет выход нейрона.

Но с помощью одного лишь перцептрона добиться хорошего результата не получится (к примеру, с его помощью невозможно решить задачу XOR). Поэтому используют многослойные нейронные сети (MLP). Веса моделей подбирают при помощи алгоритма обратного распространения ошибки. На выходе многослойной модели получается следующее значение:

$$y = f\left(\sum_{j=1}^k w_j f\left(\sum_{i=1}^n (w_{ij} x_i + b_{0j})\right) + b_0\right)$$

Исследователям удавалось успешно применить MLP для прогнозирования заболеваемости COVID-19. В [9] авторы строили прогноз для моделей с разной архитектурой. Наиболее подходящей оказалась модель, состоящая из 2 скрытых слоев. В первом слое содержится 9 нейронов, во

втором 4. Средняя квадратичная ошибка при предсказании общего количества выявленных случаев во всем мире составила $10.3 \cdot 10^6$ при горизонте прогнозирования равном 40 дней. Авторы утверждают, что модель применима как к каждой стране отдельно, так и ко всем странам одновременно.

Другой метод машинного обучения, применяемый в прогнозировании – метод опорных векторов (SVM). Это метод, основанный на статистической теории, способный решать как задачи классификации, так и регрессии.

По своей структуре SVM схож с моделями ИНС. Разница заключается в подходах к обучению. Нейронные сети обучаются за счет минимизации функции потерь методом обратного распространения ошибки. SVM же основан на принципе минимизации структурного риска, который предполагает поиск наименьшей ошибки обучения при минимизации верхней границы ошибки модели.

Общая идея понятная для случая бинарной классификации. Для заданного набора точек, принадлежащим двум классам, SVM определяет гиперплоскость так, чтобы максимальное количество точек одного класса было на одной стороне и при этом расстояние от каждого класса до плоскости максимально. При обучении модели ищется прямая линия с максимальным запасом (в общем случае гиперплоскость с оптимальным разделением). Однако на практике выборки не всегда линейно делимы. Для решения этой проблемы необходимо отобразить входные данные в многомерное пространство (пространство признаков). Этого можно добиться, используя функцию ядра. Линейные, полиномиальные и ядра с радиальной базисной функцией наиболее часто встречаются на практике.

Как уже упоминалось ранее, SVM используется как для задач классификации, так и для задачи регрессии. Для последней в литературе можно встретить аббревиатуру SVR – support vector regression. В [15] авторам удалось применить данный метод для определения заболевших коронавирусом. Разработанная модель способна предсказать поражен ли

человек COVID-19 или нет. Результаты SVM превзошли по качеству классификации такие методы как KNN, случайный лес и дерево решений. В [26] с помощью SVR исследователи предсказали ежедневное количество новых случаев в Зимбабве со средней абсолютной ошибкой 421.89 в промежутке с 20.12.2021 по 10.01.2022. Авторы утверждают, что с помощью метода опорных векторов можно получить качественный краткосрочный прогноз.

Модели, основанные на деревьях решений, также отлично подходят для прогнозирования временных рядов. Модель XGBoost одна из таких. Основу модели составляет алгоритм градиентного бустинга деревьев решений. Концепция бустинга заключается в последовательном объединении нескольких моделей прогнозирования в одну большую. Целью XGBoost является применение метода оптимизации градиентного спуска и произвольных дифференцируемых функций потерь для ее минимизации путем добавления «слабых учеников». Формально минимизацию можно записать так:

$$obj(\theta) = \sum_i L(\hat{y}_i, y_i) + \sum_k \Omega(f_k), f_k \in \mathcal{F}$$

Где L – функция потерь, измеряющая отклонение предсказанного значения \hat{y}_i от фактического y_i . Ω – функция регуляризации, которая измеряет сложность модели, предотвращая переобучение. f_k – функция из функционального пространства \mathcal{F} , являющимся множеством возможных деревьев решений. Для минимизации целевой функции XGBoost подбирает параметры для оптимальной древовидной структуры с использованием жадного алгоритма.

Данный алгоритм применяли многие исследователи при изучении распространения COVID-19. В [14] XGBoost сравнивается с SARIMA в прогнозировании количества подтвержденных случаев в США. Метрики

качества прогноза как на тренировочном наборе, так и на тестовом, лучше у XGBoost.

2.2 RNN МОДЕЛИ

Рекуррентные нейронные сети (RNN) наиболее часто встречающийся тип нейронных сетей в прогнозировании временных рядов. Их структура спроектирована таким образом, что модель способна обрабатывать информацию, учитывая историю данных. Графическое представление RNN ячейки представлена на рис. 10.

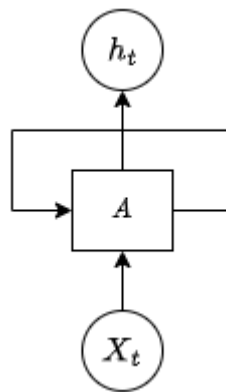


Рисунок 10. Схема RNN ячейки

На рис.10 блок «А» обозначает кусок нейронной сети. В стандартных RNN этот блок имеет простую архитектуру, например слой полносвязный слой с функцией активацией tanh.

Наличие цикла в архитектуре предоставляет возможность учитывать хронологию входных данных. По сути, рекуррентная сеть является последовательной цепочкой блоков нейронных сетей. В каждый момент времени модель получает на вход два параметра: компоненту вектора входных данных X_i и предыдущее состояние ячейки h_{i-1} .

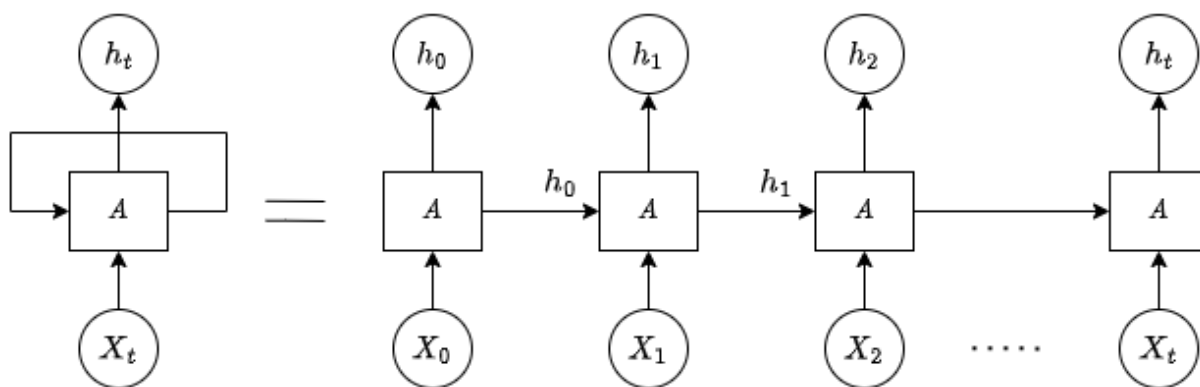


Рисунок 11. Развернутая рекуррентная нейронная сеть

Модель RNN успешно справляется со своей задачей. Многим исследователям удалось успешно применить ее во разных сферах. Однако, модель не лишена недостатков. При ее обучении градиент ошибок, проходя через каждую ячейку, становится все меньше. Из-за этого обучаются только крайние блоки модели. Веса в начальных ячейках обновляются на малую величину. Из-за этого обучение модели проходит неэффективно. Причиной этому служит большое количество слоев в архитектуре. Данную проблему решает LSTM сеть.

LSTM модель впервые опубликовали в статье [17] в 1977 году. Примечателен тот факт, что в то время ее описание оказалось слишком сложным для понимания. Поэтому LSTM модели долгое время не применяли. Архитектура разработана так, чтобы избежать проблем с затуханием градиента, которая была присуща всем RNN моделям. С их помощью были достигнуты успехи в таких сферах как машинный перевод [6], обработка изображений [19].

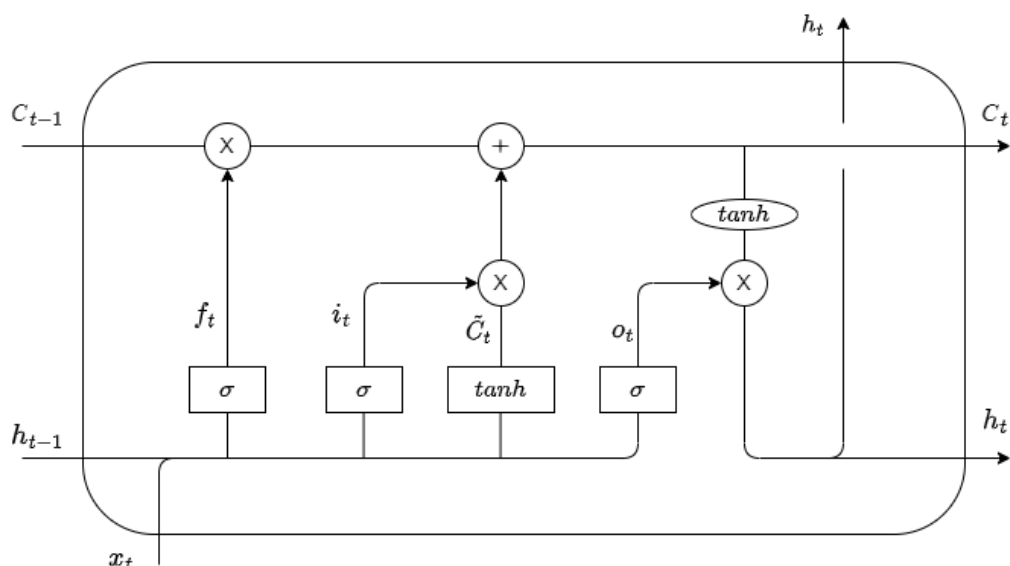


Рисунок 12. Структура LSTM ячейки

Как и другие рекуррентные модели LSTM состоит из цепочки последовательных ячеек, но вместо одного слоя нейронной сети, как это реализовано в стандартных моделях, в ячейке LSTM 4 таких слоя, взаимодействующих друг с другом особым образом. Ее структура показана выше на рис.12.

Ячейка имеет два параметра: скрытое состояние ячейки h и состояние ячейки C . Состояние ячейки похоже на конвейерную ленту. Этот параметр проходит через всю цепочку, претерпевая небольшие линейные взаимодействия.

LSTM способна удалять или добавлять новую информацию в состояние ячейки благодаря наличию фильтров. Фильтры состоят из сигмоидальной функции и операции поточечного умножения. При помощи сигмоиды регулируется количество информации, которую нужно добавить в новое состояние. В LSTM реализовано 3 таких фильтра для управления состоянием ячейки.

Первый фильтр называется «слой фильтра забывания» (forget gate layer). Он определяет какую информацию можно выбросить из состояния ячейки

исходя из значений скрытого состояния ячейки с предыдущего шага и входного значения для текущего шага:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$

Второй фильтр (input layer gate) определяет какая часть новой информации будет содержаться в состоянии ячейки. Механизм похож на предыдущий. Только добавляется слой с функцией гиперболического тангенса, который строит вектор новых значений \tilde{C}_t для добавления в состояние ячейки:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$

Теперь, когда нам известно какую часть информации из прошлого нужно забыть и какую часть из входных данных нужно добавить, можем пересчитать новое состояние ячейки:

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Новое состояние будет передаваться в последующие ячейки. Осталось решить, какую информацию нужно получать на выходе. Для этого так же, как и в других фильтрах применяем сигмоидальный слой, который покажет какую информацию o_t необходимо забрать из входных данных и скрытого состояния. Затем, пересчитанное новое состояние ячейки C_t пропускаем через \tanh слой. Полученное значение умножаем на o_t :

$$o_t = \sigma(W_o [h_{t-1}, x_t] + b_o)$$

$$h_t = o_t * \tanh(C_t)$$

Как уже говорилось ранее, с использованием LSTM удалось достичь высоких результатов во многих отраслях. Прогнозирование динамики COVID-19 не стало исключением. В [20] LSTM сети сравнивались с моделью XGBoost. При прогнозировании числа новых подтвержденных случаев на 30 дней MAPE

у LSTM модели составила 2.32% и 7.21% у XGBoost. В исследовании [18] LSTM показала лучшую точность предсказания подтвержденных случаев в Нигерии по сравнению с моделями SVM, XGBoost и наивным Байесом.

2.3 СТРАТЕГИИ ПРОГНОЗИРОВАНИЯ ВРЕМЕННЫХ РЯДОВ

Согласно [8] задача многошагового прогнозирования временных рядов заключается в предсказании H следующих значений $[y_{n+1}, \dots, y_{n+H}]$ временной последовательности $[y_1, \dots, y_n]$ состоящей из n наблюдений. H – горизонт прогнозирования. В этом разделе будут представлены основные существующие стратегии прогнозирования в машинном обучении. Далее f обозначает зависимость между наблюдениями, ε включает в себя ошибки моделирования, помехи или шум.

Рекурсивная стратегия (*Recursive*). Сначала обучается одношаговая модель f

$$y_{t+1} = f(y_t, \dots, y_{t-n+1}) + \varepsilon_{t+1}$$

Затем эта модель рекурсивно используется для получения многошагового прогноза. Основным недостатком данного метода является накопление ошибок предыдущих прогнозов. Тем не менее такая стратегия успешно используется для прогнозирования.

Прямая стратегия (*Direct*) обучает H независимых моделей f_h :

$$y_{t+h} = f_h(y_t, \dots, y_{t-n+1}) + w_{t+h}$$

После обучения результаты всех моделей объединяются и возвращается многошаговый прогноз. У данной стратегии отсутствует такой недостаток как накопление ошибки. Однако, этот метод не учитывает взаимосвязь между предсказаниями. И к тому же прямая стратегия требует большое количество времени и ресурсов.

DirRec стратегия является комбинацией двух предыдущих. *DirRec* строит прогноз для каждого значения из горизонта, используя отдельные

модели (как в технике *Direct*), но при этом добавляются значения соответствующие прогнозам предыдущего шага (как в *Recursive*).

Рекурсивные и прямые методы интервального прогнозирования имеют общую черту – все они используют модели с одним выходом. При таком подходе не сохраняется стохастическая зависимость между будущими зависимостями. Это может негативно сказаться при долгосрочном прогнозе. Исправить данный недостаток можно путем использования моделей, на выходе которых получается вектор значений. Стратегия, в которой используются модели с несколькими выходами, называется *MIMO* (Multi-Input Multi Output). Данный метод позволяет сохранить стохастическую зависимость, характеризующую временной ряд. Однако, такая стратегия ограничивает горизонт прогнозирования. Если потребуется изменить горизонт прогнозирования, то модель придется заново переобучать. Но несмотря на это, данный способ пользуется популярностью у аналитиков и исследователей.

Стратегия *DIRMO* сохраняет лучшие качества *Direct* и *MIMO* путем разделения горизонта H на несколько блоков и использования *MIMO* для прогнозирования значений внутри каждого блока. То есть для прогноза на H точек вперед потребуется $m = \frac{H}{s}$ моделей, на выходе которых s значений. Выбор оптимального параметра s позволяет сделать модель более гибкой.

2.4 ПОДГОТОВКА ДАННЫХ

Подготавливать данные для исследования будем в несколько этапов. Вначале разобьем все выборку на тренировочную, валидационную и тестовую часть в соотношении 85%, 10%, 5% соответственно. Тренировочный набор будет подаваться модели для подбора параметров. На валидационном наборе будем смотреть метрику качества предсказания. Если метрика не улучшалась более 10 эпох, то обучение прекращается и выбираются параметры с лучшей метрикой. На тестовом наборе будет проводиться оценка прогноза.

Из-за того, что ХДБ является целочисленным значением, ее динамика имеет ступенчатый вид. На таких данных модель обучается хуже. Поэтому на тренировочном наборе вычислим скользящее среднее с окном в 7 дней. Это позволит сгладить данные.

Важно, чтобы обучение нейронной сети проводилось на данных, лежащих в одном диапазоне. Не масштабируемые входные значения могут вызвать резкое изменение градиентов, что приведет к медленному или нестабильному процессу обучения. В своей работе я привожу данные к диапазону от 0 до 1 методом нормализации:

$$x_i^{scaled} = \frac{x_i - \min(X)}{\max(X) - \min(X)}$$

При обучении модели полезно будет знать информацию о частоте прогнозируемой динамики. Визуально можно сделать предположение, что у ХДБ период сезонности равен 210 дням. Чтобы как-то подтвердить эту гипотезу посмотрим на результат преобразования Фурье.

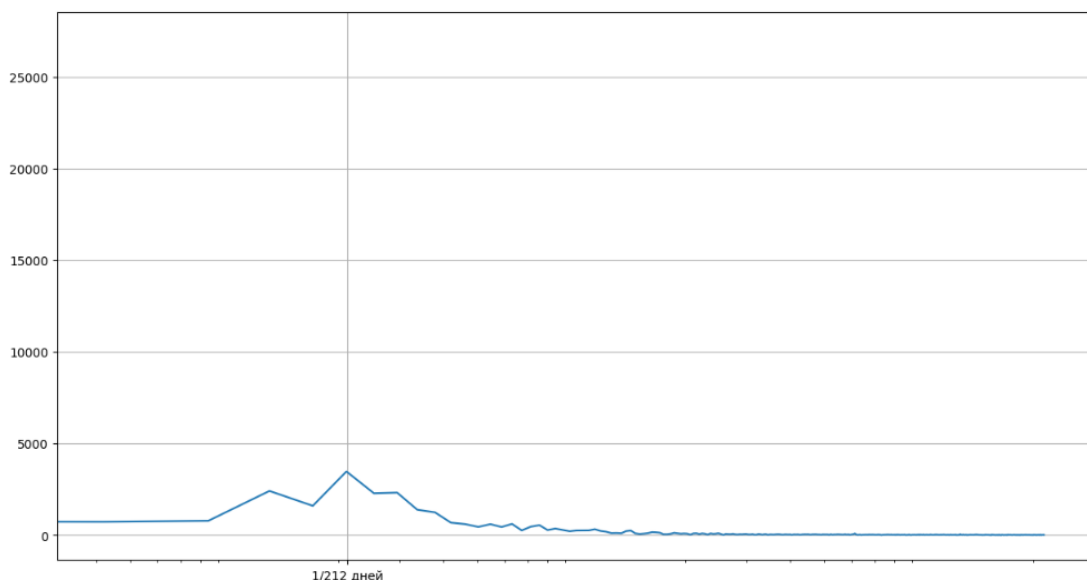


Рисунок 13. Результат быстрого преобразования Фурье

На рис.13 можем заметить небольшой скачок в районе $1/210$. Сравним как себя ведут функции $\sin\left(\frac{2\pi}{210}t\right)$ и $\cos\left(\frac{2\pi}{210}t\right)$ с исследуемой величиной:

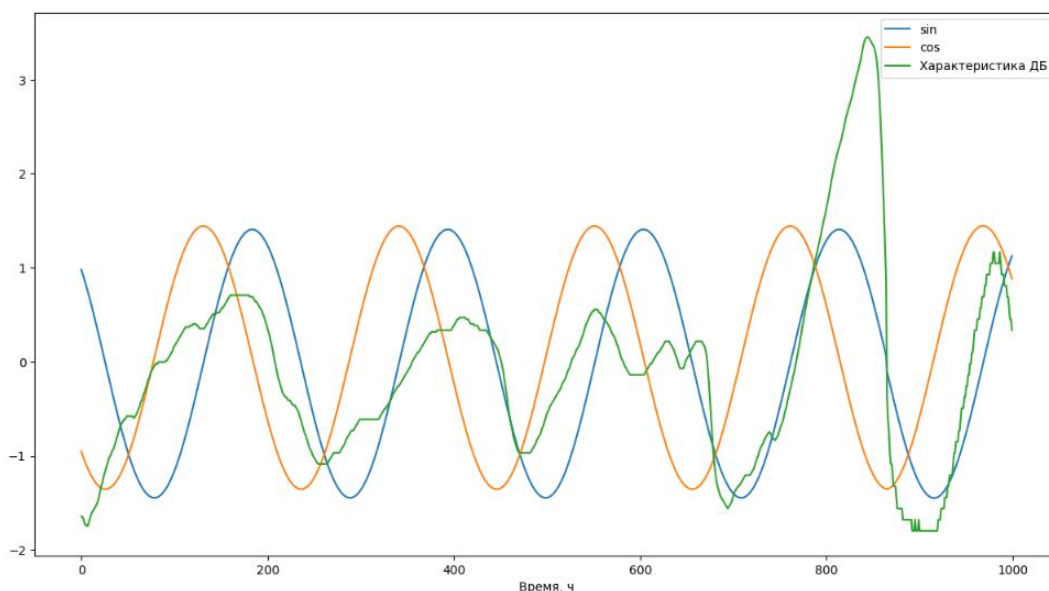


Рисунок 14. Сравнение ХДБ и компонент \sin и \cos

ХДБ и тригонометрические функции отстают на фазу. Но на мой взгляд их частоты совпадают. Компоненты тригонометрических функций будем использовать при обучении модели.

После всех преобразований модель на вход будет получать следующие параметры:

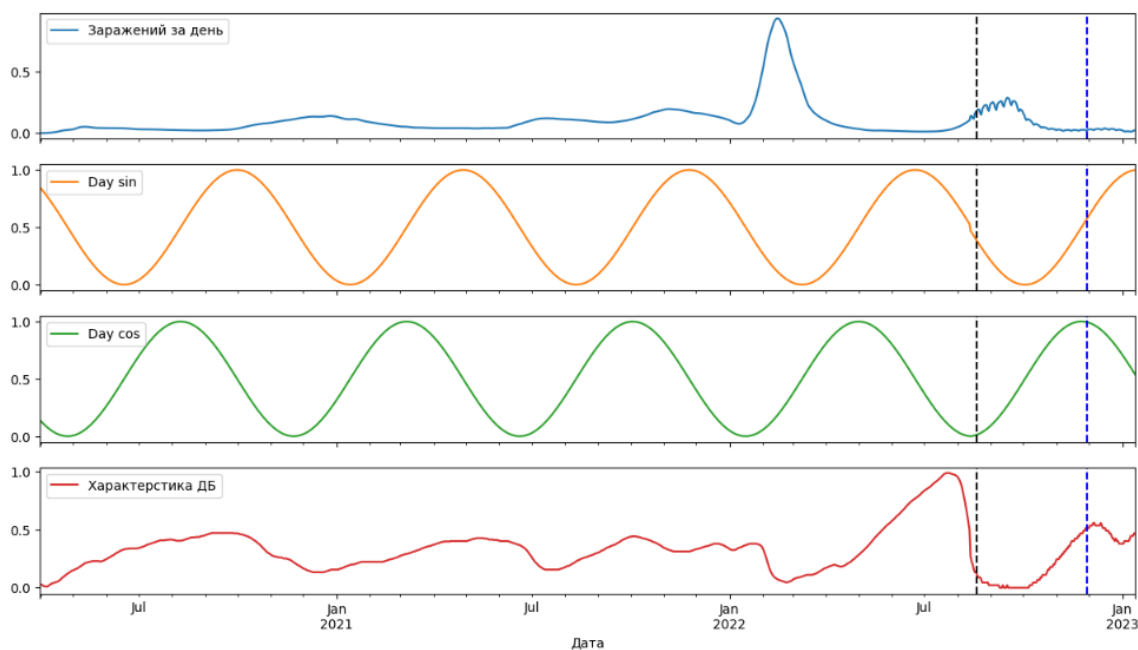


Рисунок 15. Данные участвующие в обучении

Ключевым моментом в предобработке данных является разбиение ряда на окна. Окно характеризуется следующими параметрами: ширина окна, горизонт предсказания, количество прогнозируемых величин. Как правило, ширина окна выбирается таким образом, чтобы в нем содержался один период сезонности.

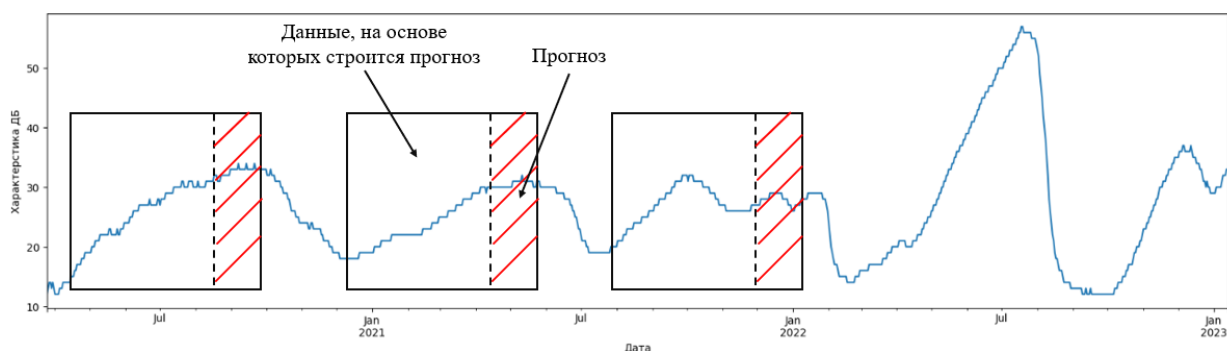


Рисунок 16. Графическое представление окон временного ряда

Затем эти окна случайным образом перемешиваются (структура ряда внутри окна сохраняется) и разбиваются на группы (batch'и) с заданным размером. Полученная структура подается на вход для обучения модели.

2.5 РЕЗУЛЬТАТЫ ПРОГНОЗИРОВАНИЯ RNN МОДЕЛЕЙ

В данном разделе будут исследоваться следующие модели: однослойные, двухслойные и трехслойные LSTM модели, Bi-LSTM, CNN-LSTM и авторегрессионная LSTM модель. Для их реализации будет использоваться библиотека TensorFlow. При выборе оптимальной модели будет изменяться размер входного окна и количество ячеек в LSTM слое. В качестве функции потерь выбрана MSE, для оптимизации обучения используется оптимизатор Adam. Прогнозировать динамику будем на 20 дней. Посмотрим, как менялась метрика MAPE и однослойной модели:

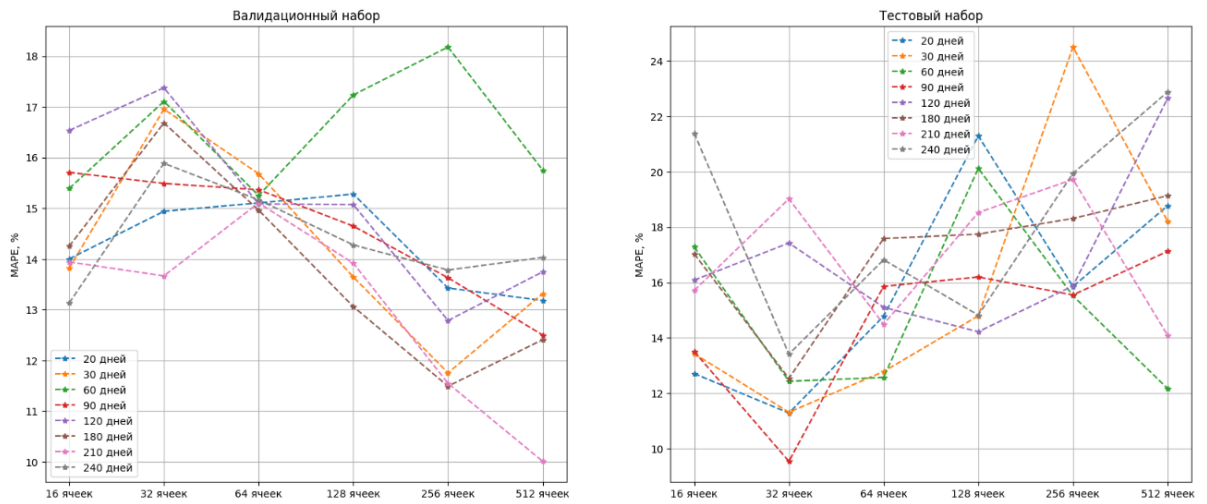


Рисунок 17. Результат подбора гиперпараметров для однослойной LSTM модели.

Однозначно определить лучшую модель сложно. Можно заметить, что на валидационном наборе с увеличением количества ячеек с 64 до 512 метрика предсказания улучшается у большинства моделей, но на тестовой выборке ошибка возрастает с увеличением ячеек. На мой взгляд, модель, состоящая из 64 ячеек с размером окна 90 дней является оптимальной. На валидационном наборе $MAPE=15.37\%$, на тестовом $MAPE=15.86\%$. Посмотрим какие результаты будут у двухслойной модели.

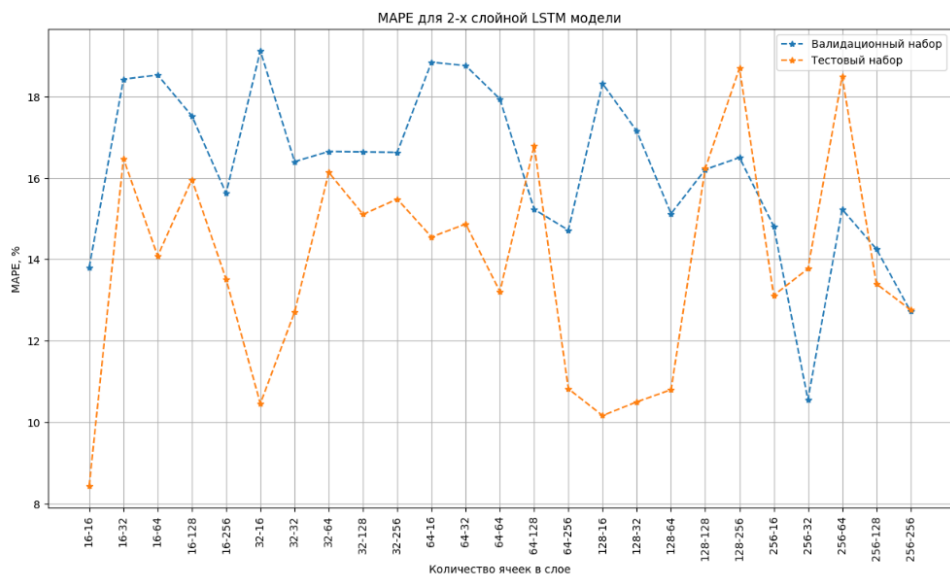


Рисунок 18. Изменение метрики предсказания при выборе количества ячеек в двухслойной LSTM модели

Как и ранее выявить явного победителя удастся с трудом. Какая-то архитектура лучше работает на валидационном наборе, какая-то на тестовом. По моему мнению, модель, имеющая в каждом слое 16 ячеек, является оптимальной для использования. Посмотрим, как меняется метрика у трехслойной модели.

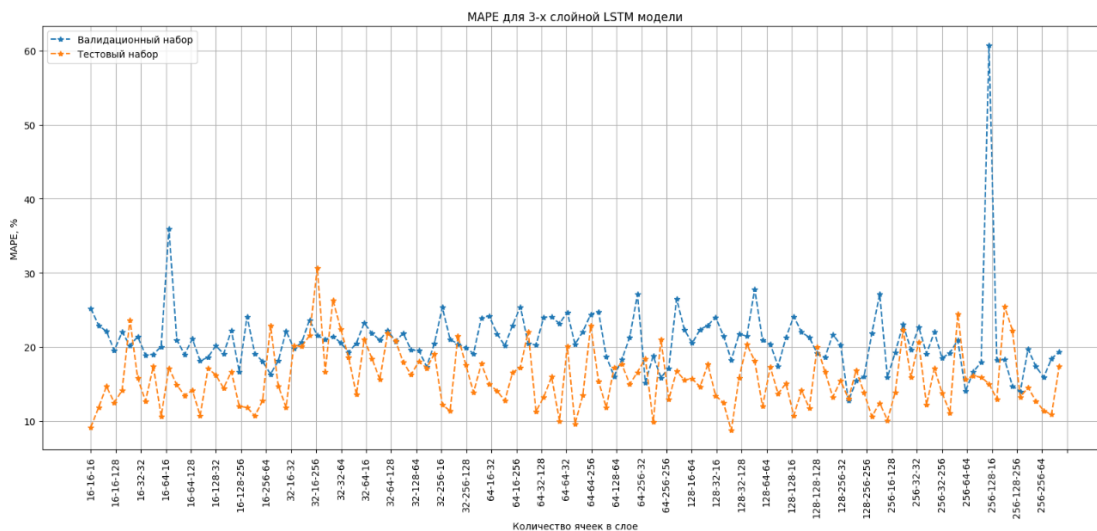


Рисунок 19. Изменение метрики при выборе количества ячеек в слое для трехслойной LSTM модели

Предсказание трехслойной модели оказалось худшим из ранее рассмотренных. Средняя MAPE на валидационном наборе составила 20%, на тестовом 15%. Многие исследователи пишут, что добавление нового LSTM слоя чаще всего негативно сказывается на результате.

Попробуем применить двунаправленную модель (Bi-LSTM) для предсказания ХДБ. В отличие от стандартной модели, информация между ячейками передается как в прямом, так и в обратном направлении. Таким образом, Bi-LSTM добавляет еще один слой LSTM, который меняет направление потока данных. Метрика прогноза получилась следующая:

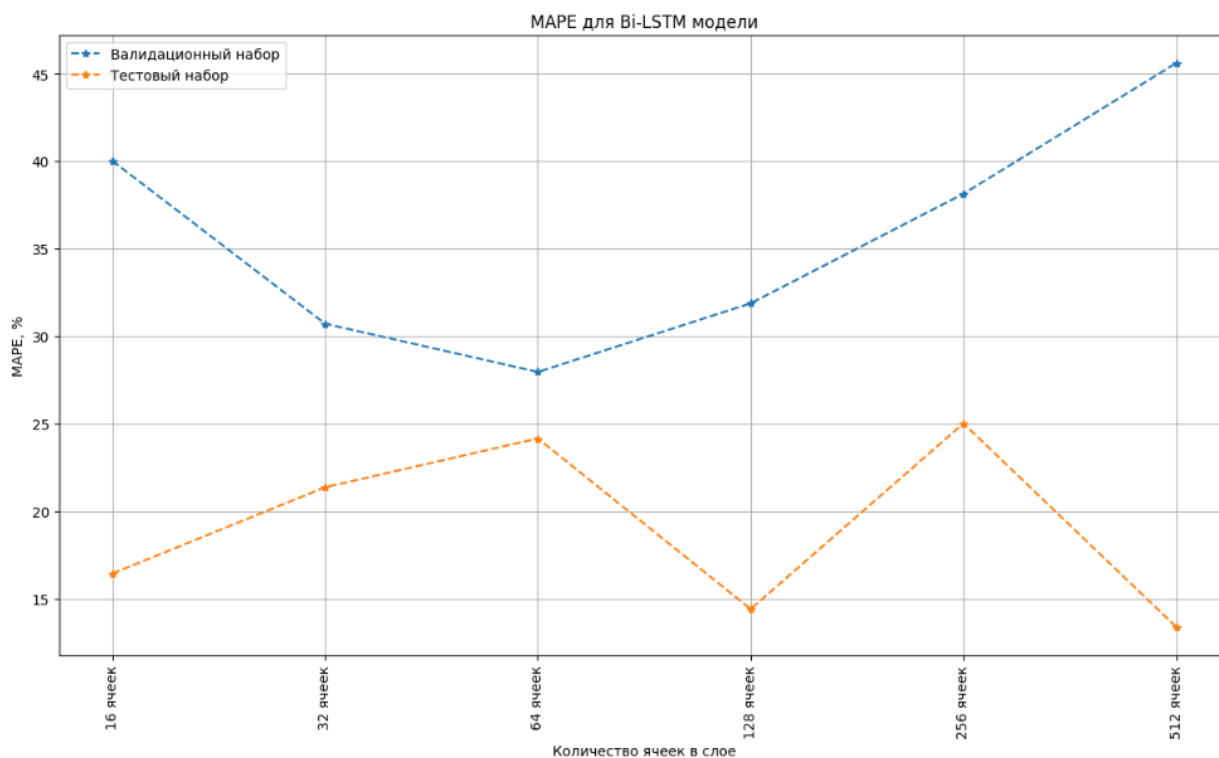


Рисунок 20. Зависимость метрики от количества ячеек для Bi-LSTM модели

Bi-LSTM предсказывает результат хуже, чем стандартная LSTM. Оптимальный результат у модели из 128 ячеек (MAPE=33% на валидационном наборе, MAPE=14.9% на тестовом)

Распространенной структурой в прогнозировании временных рядов является Encoder-Decoder. В роли декодировщика как правило выступают LSTM слои. Кодировать информацию может, как и LSTM слой, так и сверточный слой (CNN). Посмотрим, как будет меняться метрика в зависимости от размера окна. В модели используется 64 LSTM ячейки.

По результату, изображенному на рис.21, можно сделать вывод, что с увеличением размера окна качество предсказания ухудшается. К тому же, у CNN-LSTM получился результат хуже, чем у однослойной LSTM. Чтобы добиться хорошего результата для этой модели, требуется более детальное ее исследование.

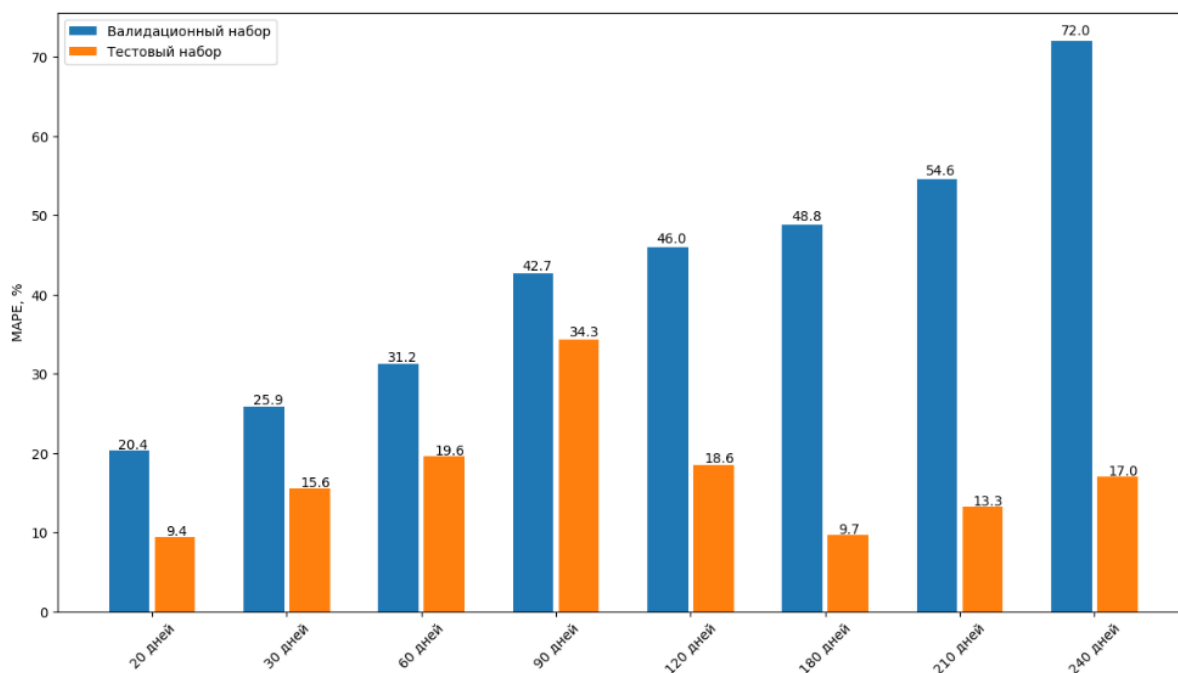


Рисунок 21. Выбор оптимального размера окна для CNN-LSTM модели.

Последняя модель, которую я рассмотрю, это авторегрессионная LSTM модель (Auto-LSTM). По сути, это та же самая однослойная модель. Разница заключается в том, что для предсказания интервала модель рекурсивно использует уже спрогнозированные значения. На рис.22 показано как меняется MAPE в зависимости от гиперпараметров. На валидационном наборе наблюдается небольшое улучшение прогноза с увеличением количества ячеек. На тестовом результат остается примерно на одном уровне. На первый взгляд кажется, что оптимальные параметры, следующие: размер окна 20 дней и 64 ячейки в слое. Мне кажется, такая модель предсказывает среднее значение динамики и совсем не улавливает тренд. Поэтому для дальнейшего исследования выберем размер окна 120 дней и 128 ячеек в слое.

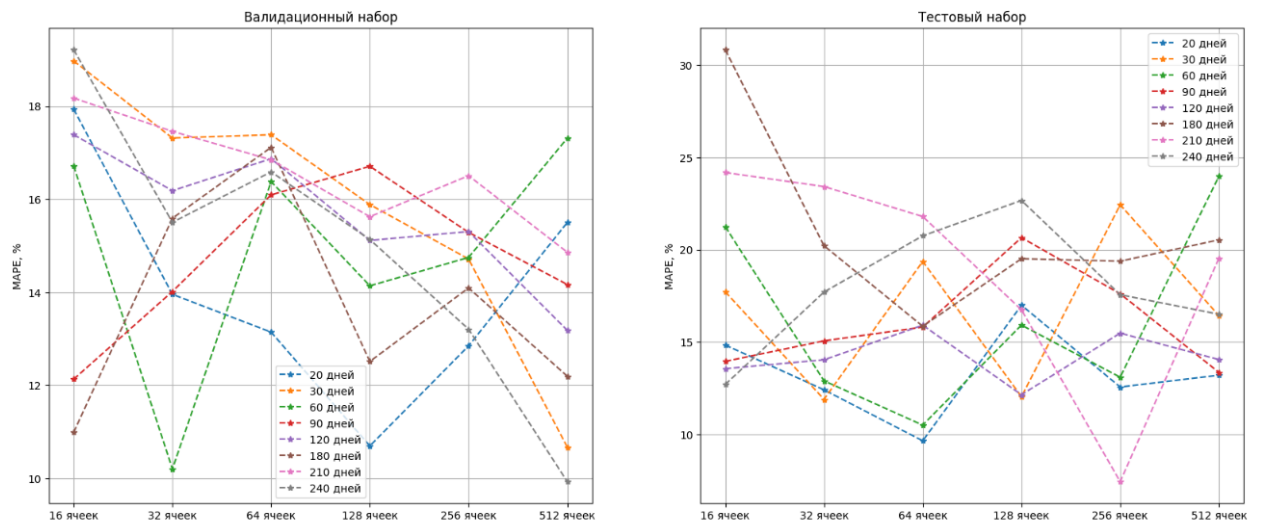


Рисунок 22. Метрика предсказания для Auto-LSTM при различных гиперпараметрах.

Оценив оптимальные параметры, увеличим количество тренировочных эпох и построим прогноз всех рассмотренных моделей.

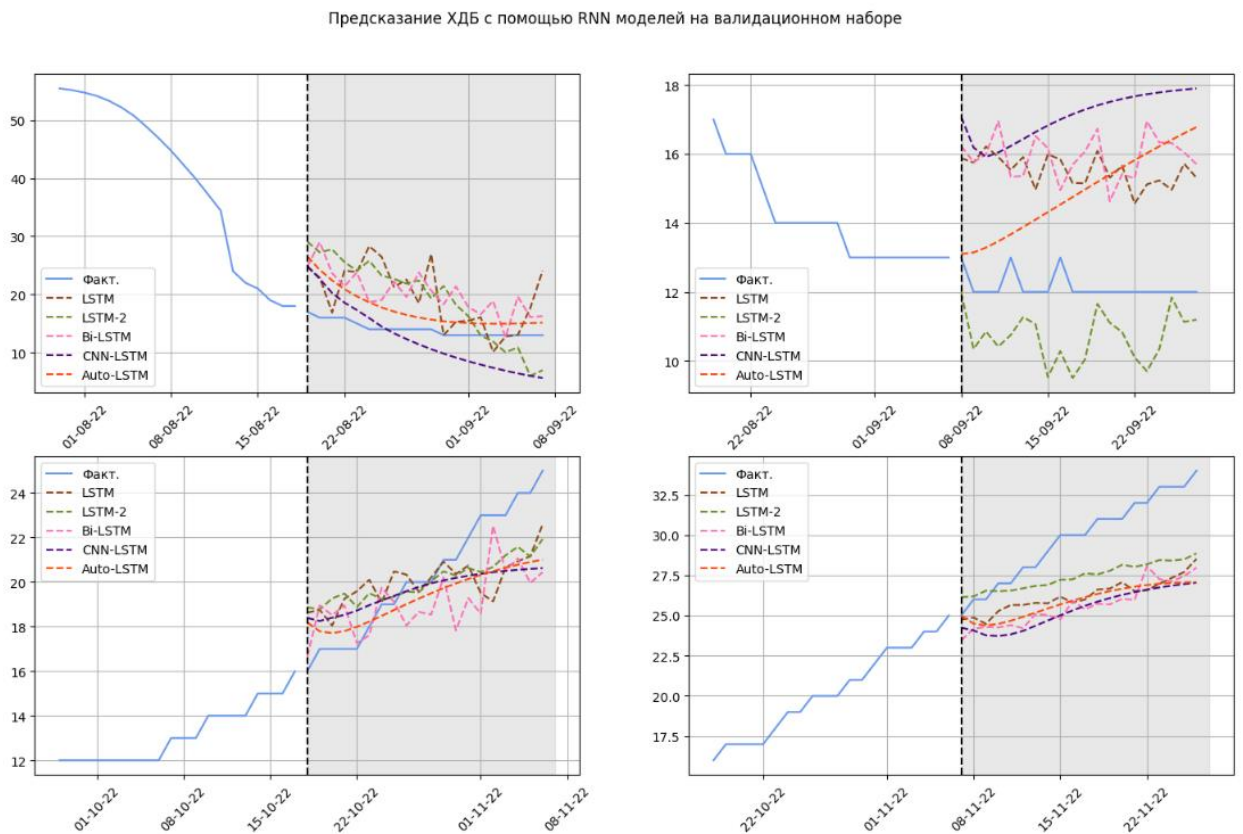


Рисунок 23. Предсказание ХДБ с помощью RNN моделей на валидационном наборе

Модель	MAPE, %	R^2
LSTM-2	13.89	-2.07
Auto-LSTM	15.79	-2.39
Bi-LSTM	17.46	-1.87
LSTM	18.25	-2.93
CNN-LSTM	23.04	-2.58

Таблица 2 Метрики прогноза на валидационном наборе для RNN моделей.

На валидационном наборе рекуррентные нейронные сети построили достаточно неплохой прогноз. Количественная мера спрогнозированных значений не сильно отличается от фактических. Но у моделей получилась очень плохая метрика R^2 . Лидером на валидационном наборе стала двухслойная LSTM модель.

Предсказания ХДБ с помощью RNN моделей на тестовом наборе

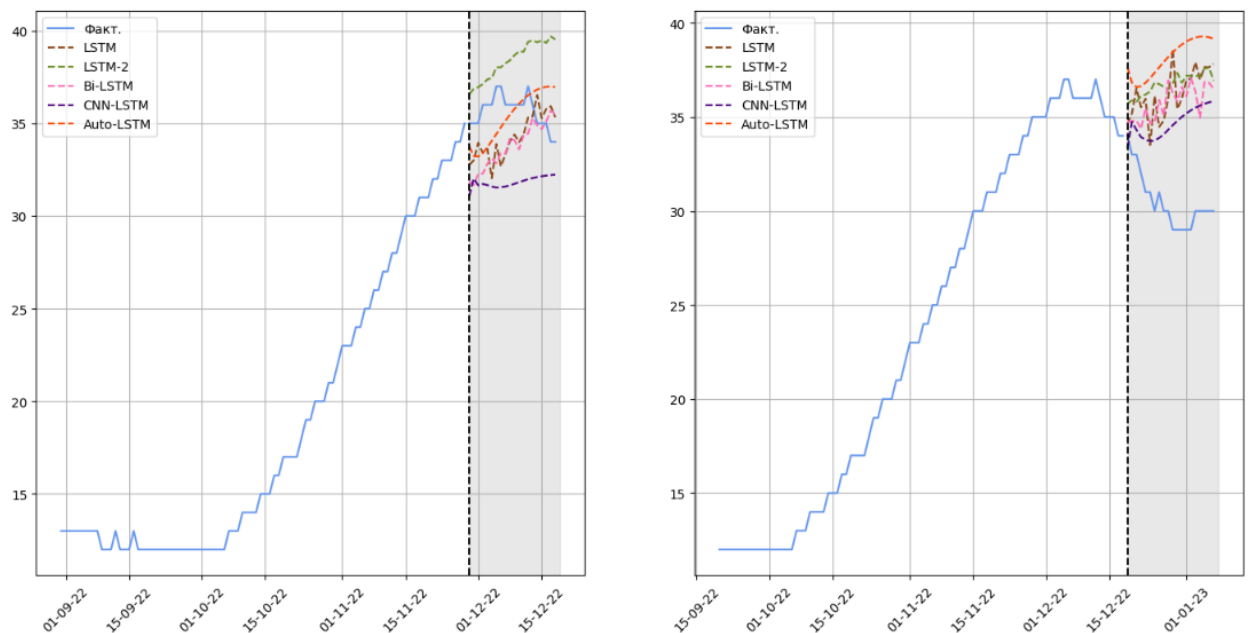


Рисунок 24. Предсказания ХДБ с помощью RNN на тестовом наборе.

Модель	MAPE, %	R^2
Bi-LSTM	12.03	-5.20
CNN-LSTM	12.62	-5.39
LSTM	12.91	-4.92

LSTM-2	17.36	-12.08
Auto-LSTM	20.55	-12.53

Таблица 3. Метрики предсказания ХДБ с помощью RNN моделей на тестовом наборе

На тестовом наборе у всех моделей уменьшилась средняя абсолютная ошибка. Но как мы можем заметить и по столбцу с R^2 , и визуально на рис.24, у всех моделей не получилось предсказать тренд динамики.

Резюмируя итог данной главы, можно сделать следующий вывод. Рекуррентные нейронные сети способны справиться с прогнозированием временных рядов. MAPE у RNN оказалось меньше чем у статистических моделей. Правда с их помощью не удалось точно предсказать тренд динамики.

ГЛАВА 3. DA RNN МОДЕЛЬ

3.1 ОБЗОР МОДЕЛЕЙ CODER-DECODER С МЕХАНИЗМОМ ВНИМАНИЯ

Используя базовые реализации рекуррентных нейронных сетей, не удалось получить качественного предсказания. У всех моделей не получается предсказать тренд динамики. Необходимо усложнять архитектуру сети.

Одной из популярных структур считаются сети кодировщик-декодировщик (Encoder-Decoder). Они получили известность благодаря своему успеху в машинном переводе. Их идея кроется в названии: входные данные кодируются в вектор фиксированной длины, затем используется блок декодировщика для получения финального результата. Для такой структуры обычно очень сложно подобрать гиперпараметры. Часто при увеличении размера входного окна, качество модели ухудшается. В этом можно убедиться, посмотрев на результаты модели CNN-LSTM во второй главе.

Данную проблему решают сети кодировщика-декодировщика на основе механизма внимания. Впервые такое решение было представлено в [6]. Концепция этого механизма основана на восприятии информации человеком. К примеру, на более насыщенные детали с высокой контрастностью человеческий глаз обращает свое внимание в первую очередь. И что самое важное, мозг дольше сохраняет в памяти ту информацию, на которую он больше всего обратил внимание.

В нейронных сетях механизм внимания динамически выделяет более заметную часть информации подобно человеческому мозгу. Он состоит из 3 компонентов:

1. Компонента, считывающая сырые данные и преобразующая их в распределенный набор значений для одного признака.
2. Компонента, вычисляющая весовые коэффициенты для каждого наблюдения.
3. Компонента, в которой высчитывается взвешенная сумма значений.

Нейронные сети типа кодировщик-декодировщик, основанные на механизме внимания, показали высокую результативность при решении задач машинного перевода, обработки изображений. Однако, как утверждается в [25], они могут не показать такой же результат в прогнозировании временных рядов. Поэтому авторы предложили свою модель DA RNN с двухэтапным механизмом внимания. Архитектура сети представлена на рис. 25

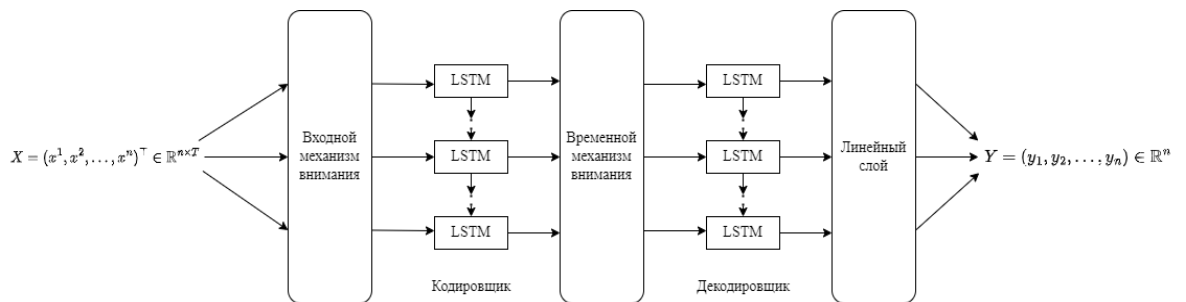


Рисунок 25. Архитектура сети DA RNN

В модели можно выделить два блока: блок кодировщика с входным механизмом внимания и блок декодировщика с временным механизмом внимания. Рассмотрим отдельно каждый блок.

3.2 ВХОДНОЙ МЕХАНИЗМ ВНИМАНИЯ

DA RNN состоит из двух частей. Первая часть состоит из кодировщика с входным слоем внимания. Его задача адаптивно извлекать полезную информацию из ряда на каждом временном шаге. На рис.26 представлена схема вычисления взвешенного значения в момент времени t .

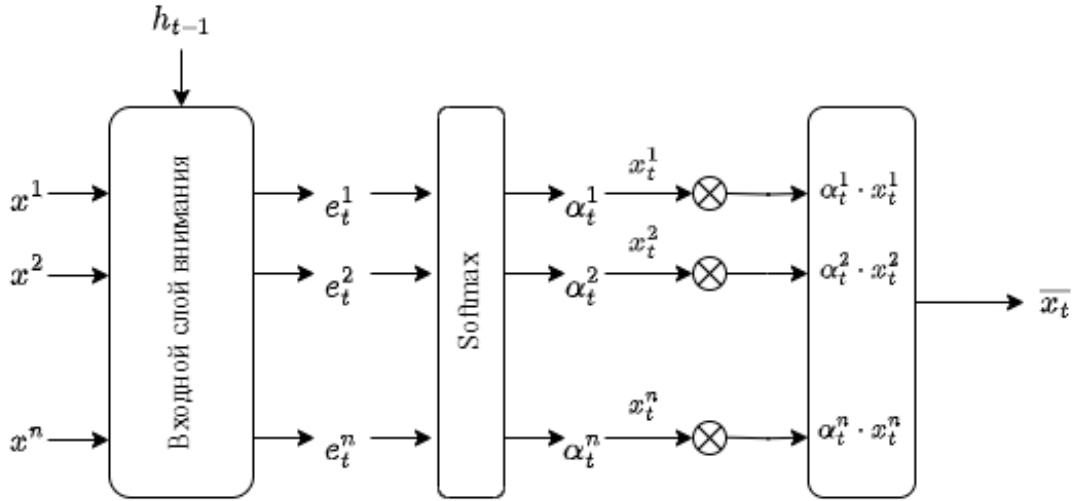


Рисунок 26. Входной механизм внимания

Где x^k – k -ое окно временного ряда, h_{t-1} – скрытое состояние ячейки для момента времени $t-1$.

Рассмотрим, как устроен этот механизм на примере x^k окна временного ряда. В каждый момент времени на вход в слой внимания (по сути, это обычный полносвязный слой) подается целое окно, скрытое состояние ячейки $h_{t-1} \in \mathbb{R}^m$ и состояние ячейки $s_{t-1} \in \mathbb{R}^m$ с предыдущего шага

$$e_t^k = v_e^\top \tanh(W_e[h_{t-1}; s_{t-1}] + U_e x^k)$$

Где $v_e \in \mathbb{R}^T$, $W_e \in \mathbb{R}^{T \times 2m}$, $U_e \in \mathbb{R}^{T \times T}$ – параметры для обучения, m – количество ячеек в кодировщике

Подсчитанные значения передаются в функцию softmax для того, чтобы сумма e_t^k равнялась 1:

$$\alpha_t^k = \frac{\exp(e_t^k)}{\sum_{i=1}^n \exp(e_t^i)}$$

Вычислив весовые коэффициенты внимания, получаем вектор взвешенных наблюдений:

$$\tilde{x}_t = (\alpha_t^1 x_t^1, \alpha_t^2 x_t^2, \dots, \alpha_t^n x_t^n)^\top$$

Используя этот вектор, обновляем скрытое состояние для момента времени t :

$$h_t = f_1(h_{t-1}, s_{t-1}, \tilde{x}_t)$$

Где f_1 – LSTM ячейка кодировщика.

3.3 БЛОК DECODER С ВРЕМЕННЫМ МЕХАНИЗМОМ ВНИМАНИЯ

Блок Decoder используется для расшифровки закодированной информации от блока Encoder. Для того, чтобы предотвратить ухудшение результата с увеличением входной последовательности, в этом блоке также используется механизм внимания. Графическое представление данного процесса представлено на рис.27.

Веса внимания вычисляются на основе состояния кодировщика h_i , скрытого состояния $d_{t-1} \in \mathbb{R}^p$ и состояния $s'_{t-1} \in \mathbb{R}^p$ LSTM ячейки декодировщика:

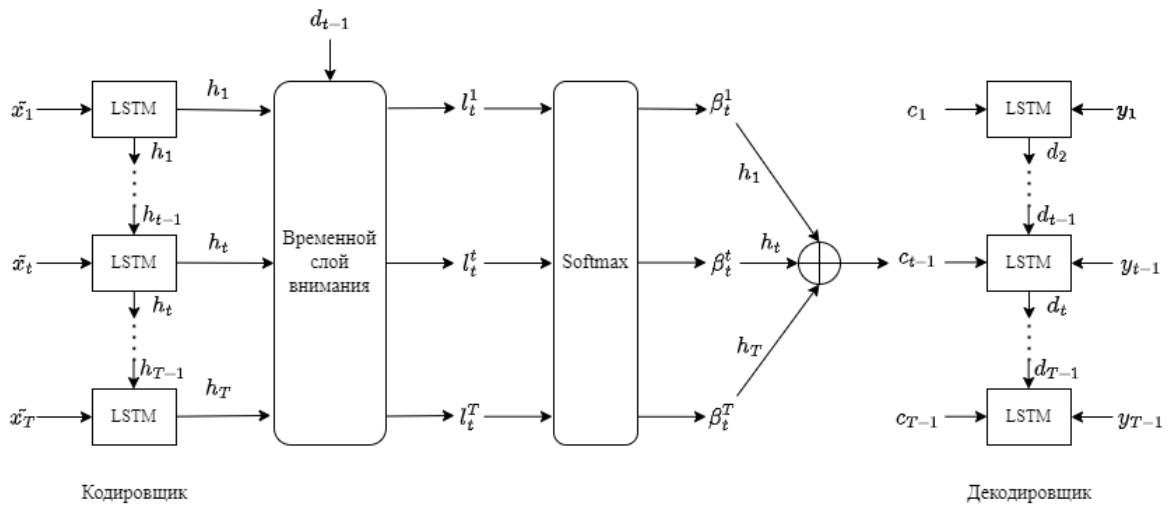


Рисунок 27. Временной механизм внимания

$$l_t^i = v_d^\top \tanh(W_d[d_{t-1}; s'_{t-1}] + U_d h_i), \quad 1 \leq i \leq T$$

$$\beta_t^i = \frac{\exp(l_t^i)}{\sum_{j=1}^T \exp(l_t^j)}$$

Где $v_d \in \mathbb{R}^m$, $W_d \in \mathbb{R}^{m \times 2p}$, $U_d \in \mathbb{R}^{m \times m}$ – параметры для обучения, p – количество ячеек в декодерошке.

Далее вычисляется взвешенная сумма скрытых состояний кодировщика h_i :

$$c_t = \sum_{i=1}^T \beta_t^i h_i$$

В этом и заключается отличие временного механизма внимания от входного. Вычисленный вектор контекста c_t разный на каждом моменте времени.

После того, как мы получили вектор контекста объединяем его с вектором целевых значений и отправляем их в линейный слой:

$$\tilde{y}_{t-1} = \tilde{w}^\top [y_{t-1}; c_{t-1}] + \tilde{b}$$

Где $\tilde{w} \in \mathbb{R}^{m+1}$, $b \in \mathbb{R}$ параметры для обучения.

Новое вычисленное значение \tilde{y}_{t-1} используем для обновления скрытого состояния декодерошки:

$$d_t = f_2(d_{t-1}, s'_{t-1}, \tilde{y}_{t-1})$$

3.4 РЕЗУЛЬТАТЫ ПРОГНОЗИРОВАНИЯ МОДЕЛИ DA RNN

В обучении модели используются такие же инструменты, что и у ранее рассмотренных. Модель реализована с помощью библиотеки Tensorflow. В качестве функции потерь рассматривается средняя квадратичная ошибка, для оптимизации обучения используется оптимизатор Adam. Отличие заключается в том, что при обучении модель предсказывает одну точку. Чтобы получить значения для всего горизонта предсказания, предсказанные значения для момента времени t используются как входные для следующего момента времени $t + 1$. То есть используется рекурсивный метод прогнозирования.

Из-за технических особенностей реализации модели вычисления нельзя перенести на графический ускоритель GPU. Поэтому при оценке гиперпараметров ограничимся шириной окна.

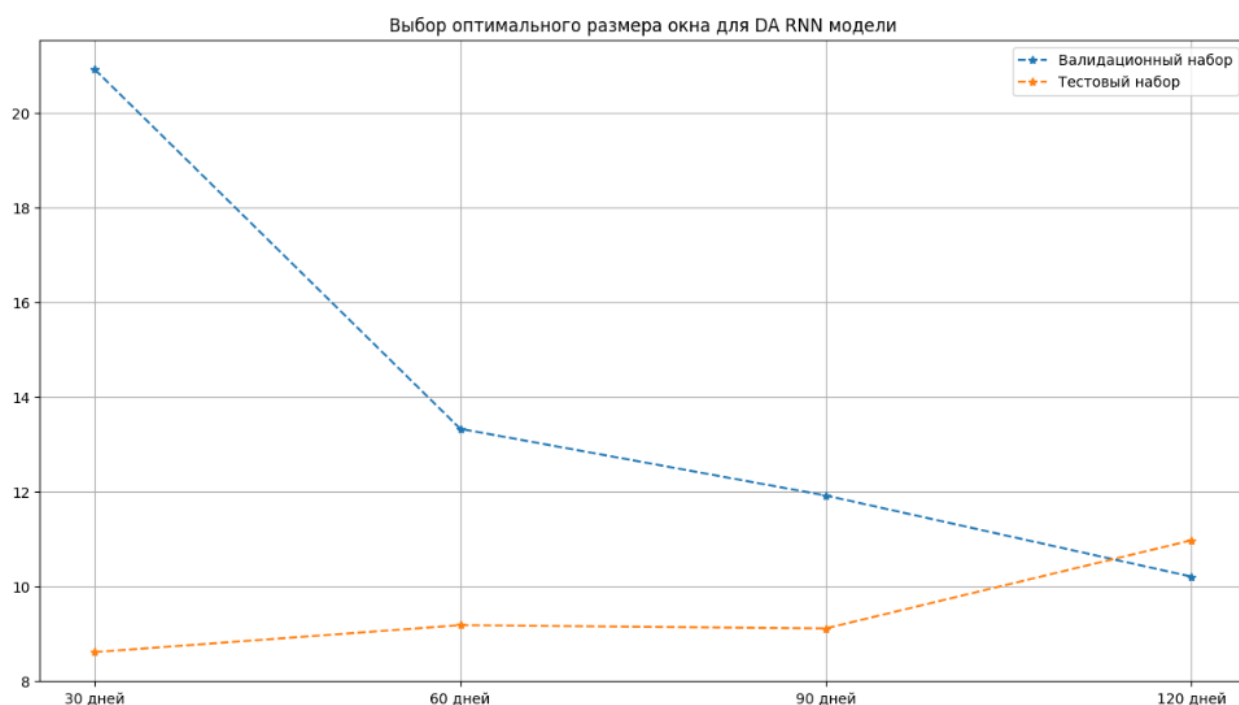


Рисунок 28. Выбор оптимального размера окна для DA RNN модели

На рис.28 можно заметить, что с увеличением размера окна уменьшается ошибка прогноза на валидационной выборке, и наоборот, увеличивается на тестовой. Визуально, ширина окна равная 90 дням является оптимальным значением. Посмотрим какой прогноз удастся построить, если в слоях Encoder и Decoder будет по 64 LSTM ячейки.

Модель	MAPE, %	R ²
LSTM-2	13.89	-2.07
DA-RNN	14.66	-1.95
Auto-LSTM	15.79	-2.39
Bi-LSTM	17.46	-1.87
LSTM	18.25	-2.93
CNN-LSTM	23.04	-2.58

Таблица 4. Метрика предсказания на валидационном наборе.

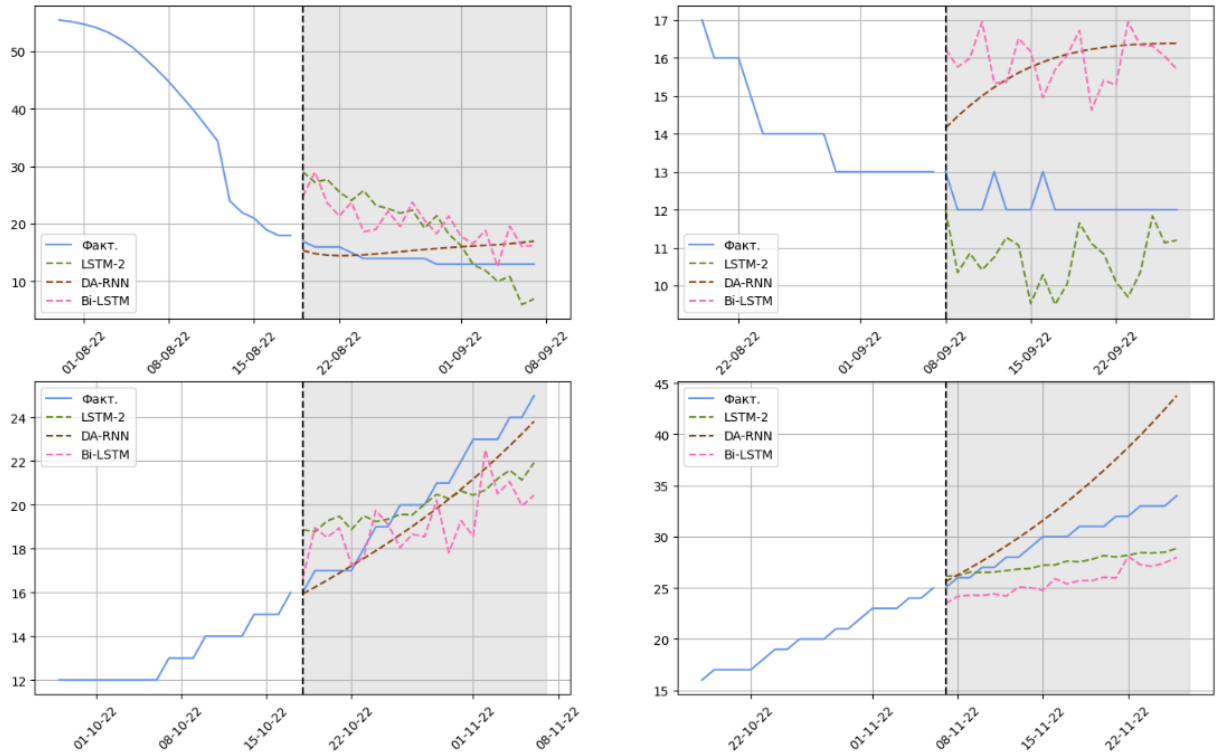


Рисунок 29. Предсказание модели DA-RNN на валидационном наборе

На валидационном наборе не удалось улучшить прогноз. Это можно заметить как визуально, так и по метрикам качества. Но все равно, модель показала неплохой результат. Построим прогноз на тестовой выборке.

Модель	MAPE, %	R ²
DA-RNN	6.23	-0.77
Bi-LSTM	12.03	-5.20
CNN-LSTM	12.63	-5.39
LSTM	12.91	-4.92
LSTM-2	17.37	-12.08
Auto-LSTM	20.55	-12.53

Таблица 5. Метрика предсказания на тестовом наборе.

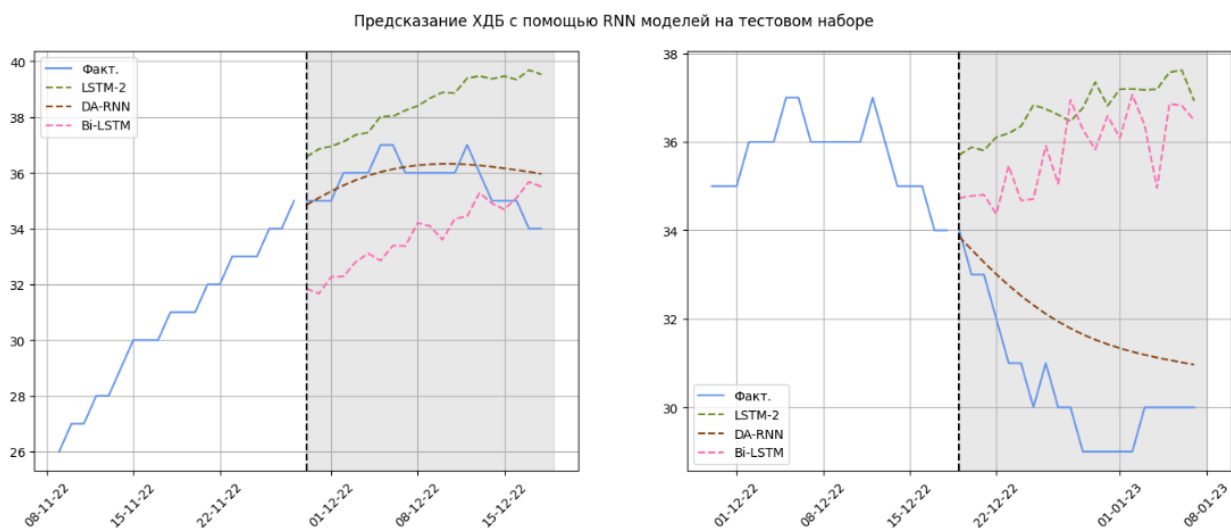


Рисунок 30. Предсказание модели DA-RNN на тестовой выборке

На тестовой выборке уже наглядно видно, что DA-RNN прогнозирует лучше рассмотренных моделей. Это подтверждается и в таблице 5.

Рассмотрим, как модель себя проявит в условиях отсутствия большого количества информации. Ограничим обучающую выборку периодом равным продолжительности первой волны (с 25.03.2020 по 16.01.2021). Необходимо предсказать поведение второй волны.

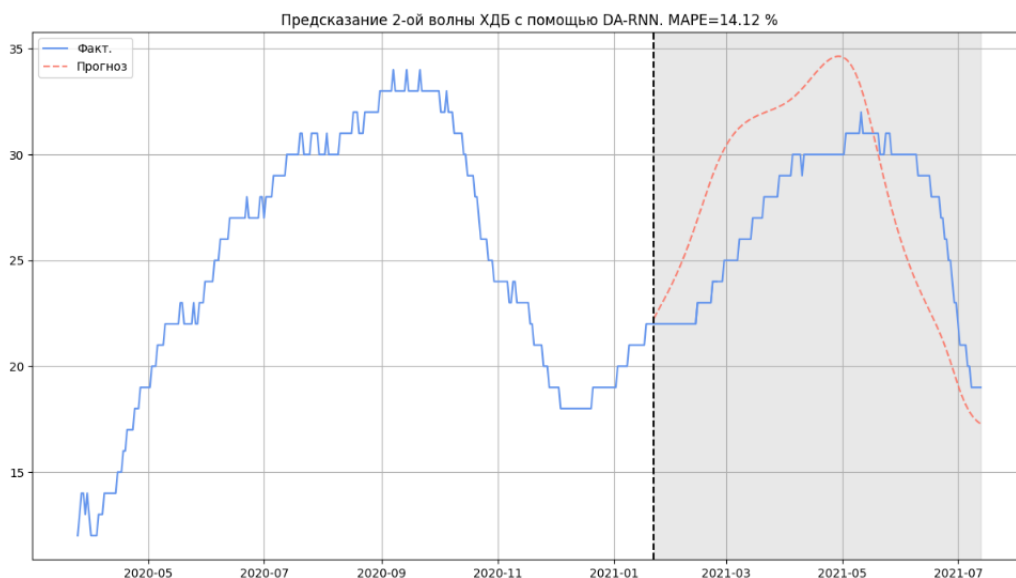


Рисунок 31. Предсказание 2-й волны ХДБ с помощью DA-RNN

По рис. 31 можно сделать вывод, что DA-RNN хорошо справилась с предсказанием 2-й волны характеристики динамического баланса. У

прогнозных значений низкая средняя абсолютная ошибка (MAPE=14.12%). Также удалось поймать тренд динамики. Посмотрим, как модель справится с предсказанием 3-й волны:

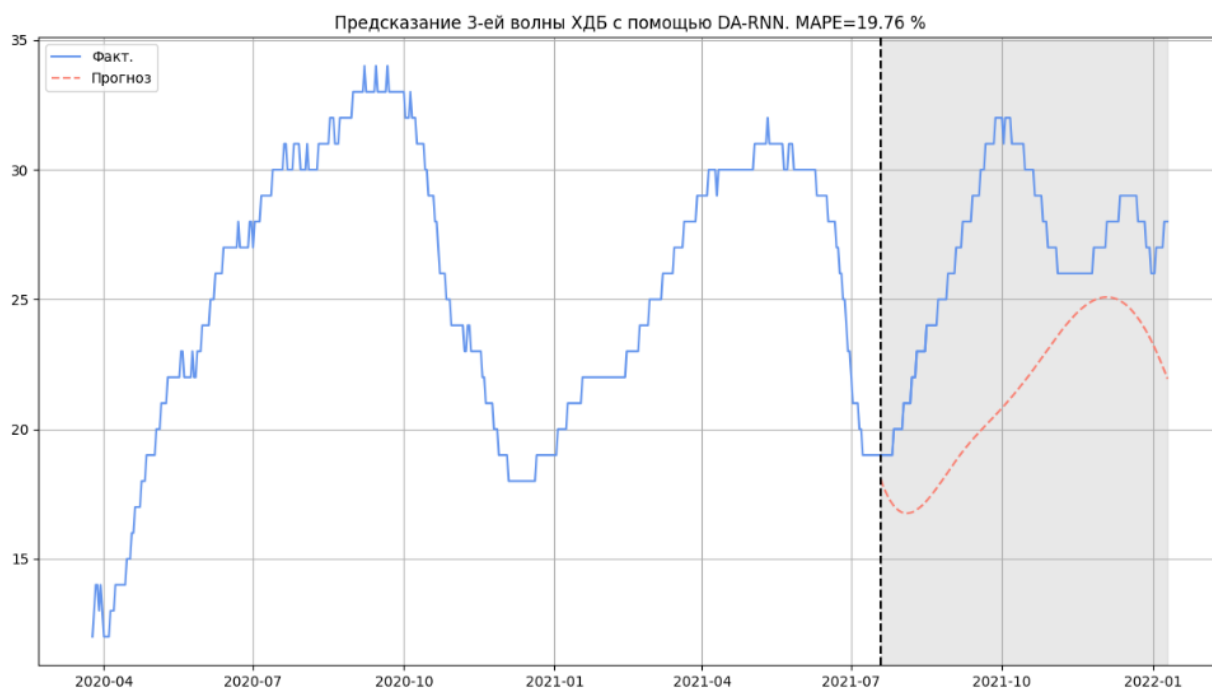


Рисунок 32. Предсказание 3-й волны ХДБ с помощью DA-RNN

Качество прогноза 3-й волны ХДБ хуже, чем у прошлой волны. Ошибка увеличилась на 5%, а также тренд предсказания плохо совпадает с истинным.

В данной главе было проведено исследование модели, основанной на двухэтапном механизме внимания (DA RNN). Ошибку прогноза удалось уменьшить на 6.2% на тестовом наборе. Модель значительно лучше предсказывает тренд динамики на тестовой выборке. В следующей главе будет проведена оценка количества болеющих людей на основе сделанного прогноза.

ГЛАВА 4. ОЦЕНКА КОЛИЧЕСТВА ИНФИЦИРОВАННЫХ ЛЮДЕЙ

4.1 МОДЕЛЬ СВRR

Метод прецедентов для прогнозирования процентного прироста (Case-Based Rate Reasoning) был предложен в работах [2, 29]. С его помощью можно эвристически оценить значение процентного прироста, основываясь на динамике стран-лидеров, и общее количество подтверждённых случаев заболеваний COVID-19. В своих работах авторам удалось предсказать исследуемую величину с высокой точностью. Стандартное отклонение за период с 20.04.2020 по 31.07.2020 составило 3.7%.

Итак, рассмотрим из чего состоит данный метод. Общее количество подтвержденных случаев $C(t)$ можно представить в виде:

$$C(t) = \left(1 + \frac{r(t)}{100}\right) \times C(t - 1)$$

Где $r(t)$ – процентный прирост новых подтвержденных случаев за день.

Исследуемый горизонт развития пандемии разбивается на M промежутков:

$$(T_{m-1}, T_m], \quad 0 \leq T_{m-1} < T_m \leq T, \quad m = 1, 2, \dots, M$$

Для любого $k = 1, 2, \dots, (T_m - T_{m-1})$ справедливо следующее выражение:

$$C(T_{m-1} + k) = \left(1 + \frac{r(T_{m-1} + k)}{100}\right) \times C(T_{m-1} + k - 1),$$
$$k = 1, 2, \dots, (T_m - T_{m-1})$$

Тогда значение общего числа подтвержденных случаев на конце промежутка вычисляется:

$$C(T_m) = \left(1 + \frac{r(T_{m-1} + 1)}{100}\right) \times \left(1 + \frac{r(T_{m-1} + 2)}{100}\right) \times \dots \times \left(1 + \frac{r(T_m)}{100}\right) \\ \times C(T_{m-1})$$

В CBRR для построения прогноза во всем рассматриваемом интервале $(T_{m-1}, T_m]$ необходимо оценить процентный прирост $\tilde{r}(T_{m-1} + k)$. Авторы предлагают оценить процентный прирост следующим выражением:

$$\tilde{r}(T_{m-1} + k) = \tilde{r}(T_{m-1} + k - 1) - A^{(k)}(\tilde{r}(T_{m-1}) - \tilde{r}_m), \\ k = 1, 2 \dots, (T_m - T_{m-1})$$

Где \tilde{r}_m – фиксированный уровень процентного прироста, выбранный на основе динамики страны-предшественника. Для России такими странами считаются Италия, Испания, Великобритания и Франция.

$A^{(k)}$ могут быть случайные или детерминированные числа из промежутка $[0; 1]$, которые удовлетворяют условию:

$$\sum_{k=1}^{T_m - T_{m-1}} A^{(k)} = 1$$

Авторы воспользовались равномерным распределением для вычисления коэффициентов $A^{(k)}$:

$$A^{(k)} = \frac{1}{T_m - T_{m-1}}, \quad k = 1, 2 \dots, (T_m - T_{m-1})$$

Используя данный метод, построим прогноз для 2-х интервалов. Первый интервал соответствует долгосрочному прогнозу, в котором происходит вторая волна роста ХДБ с 22.01.2021 по 27.07.2021. Во втором сделаем краткосрочный прогноз на 20 дней в период с 29.11.2022 по 17.12.2022.

Оценим процентный прирост по стране-предшественнику (Испании). На рис. 33 и 34 представлены графики с динамикой. Можно заметить, что характер динамики в Испании опережает динамику в России.

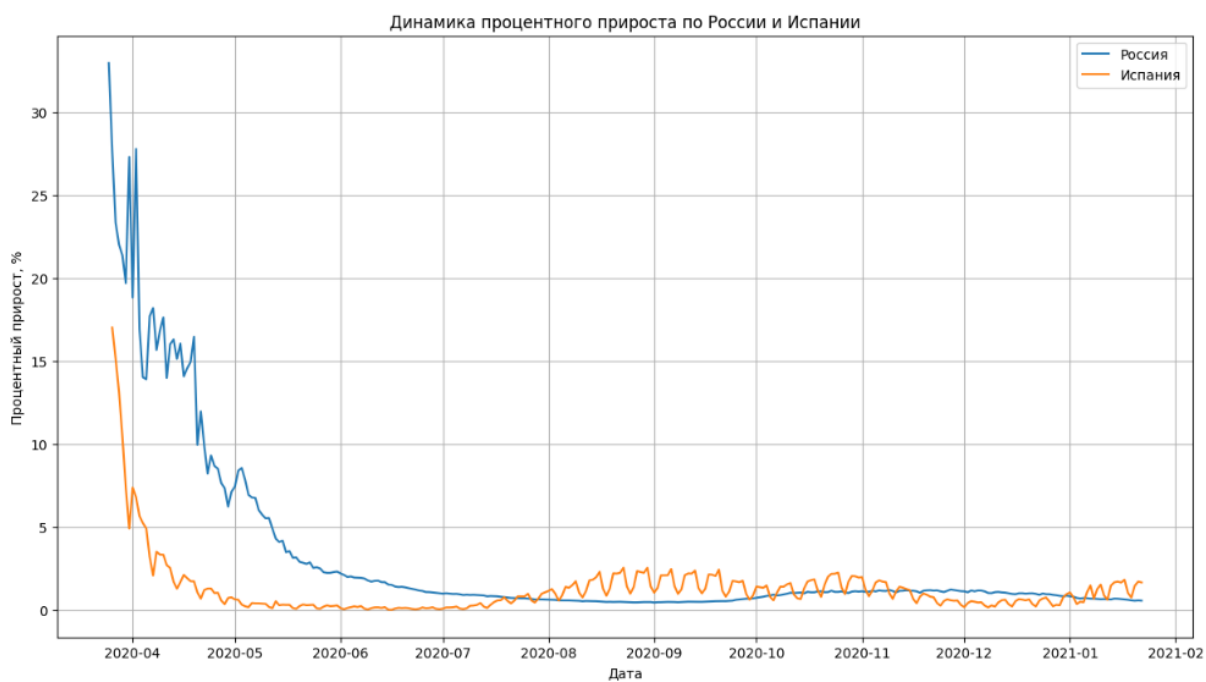


Рисунок 33. Динамика процентного прироста по России и Испании

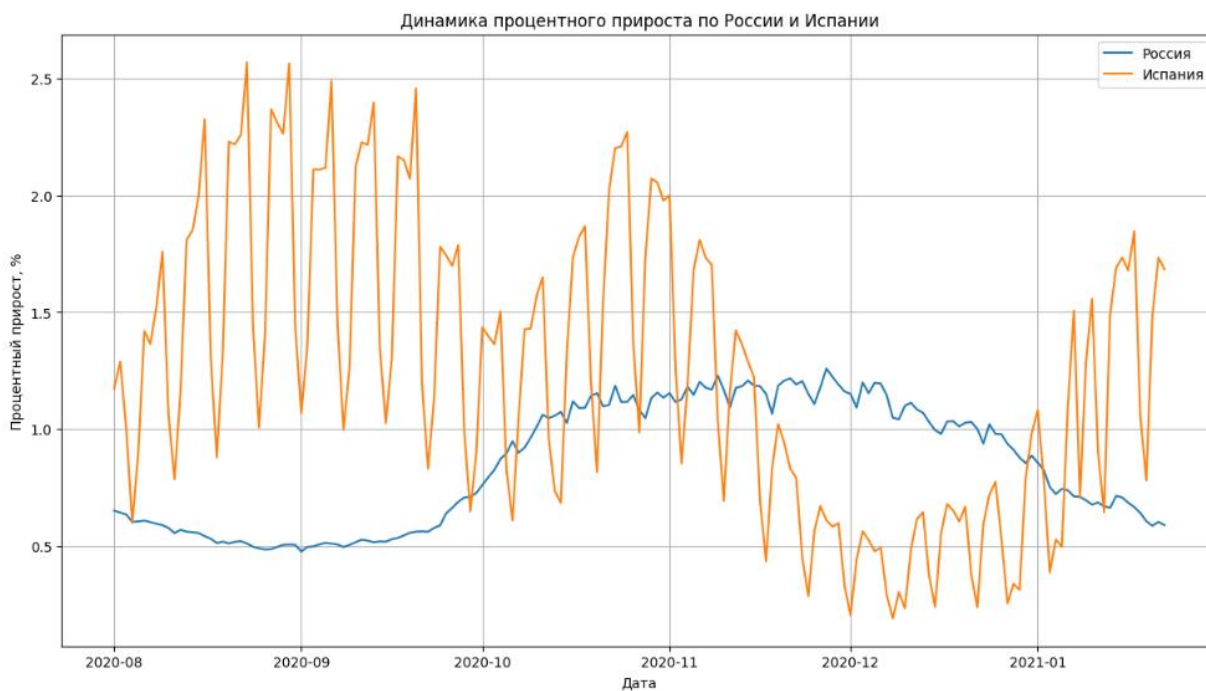


Рисунок 34. Динамика процентного прироста по России и Испании
(детальная).

Основываясь на динамике страны-лидера, сделаем предположение, что в следующий момент времени в России будет спад процентного прироста до 0.2 %, а затем будет рост до 0.5%.

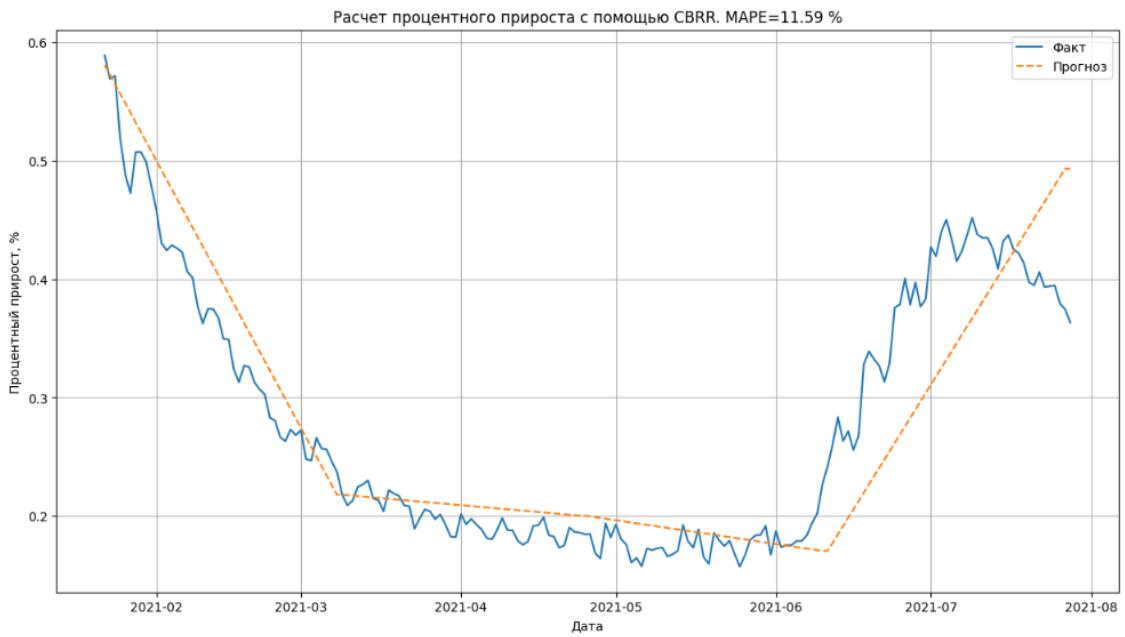


Рисунок 35. Расчет процентного прироста для периода с 22.01.2021 по 27.07.2021

По такому же принципу оценим будущие значения прироста для второго промежутка. С учетом количественной амплитуды прироста в России и поведения тренда динамики в Италии примем следующие пороговые значения: 0.3%, 0.25%, 0.4%. Результат прогноза представлен на рис. 37

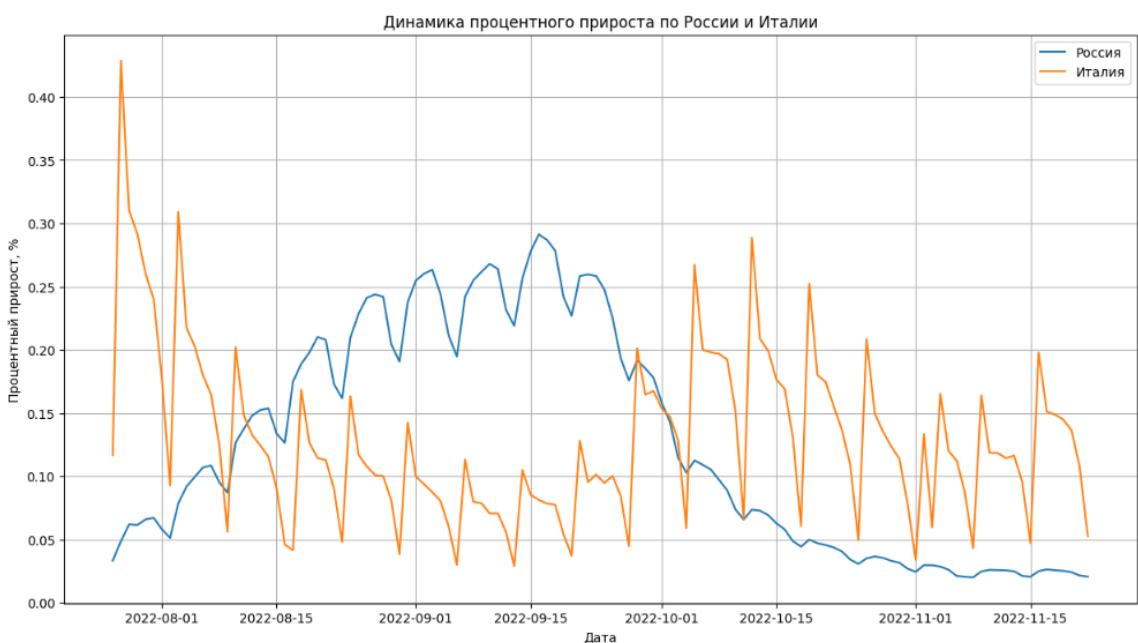


Рисунок 36. Динамика процентного прироста в России и Италии

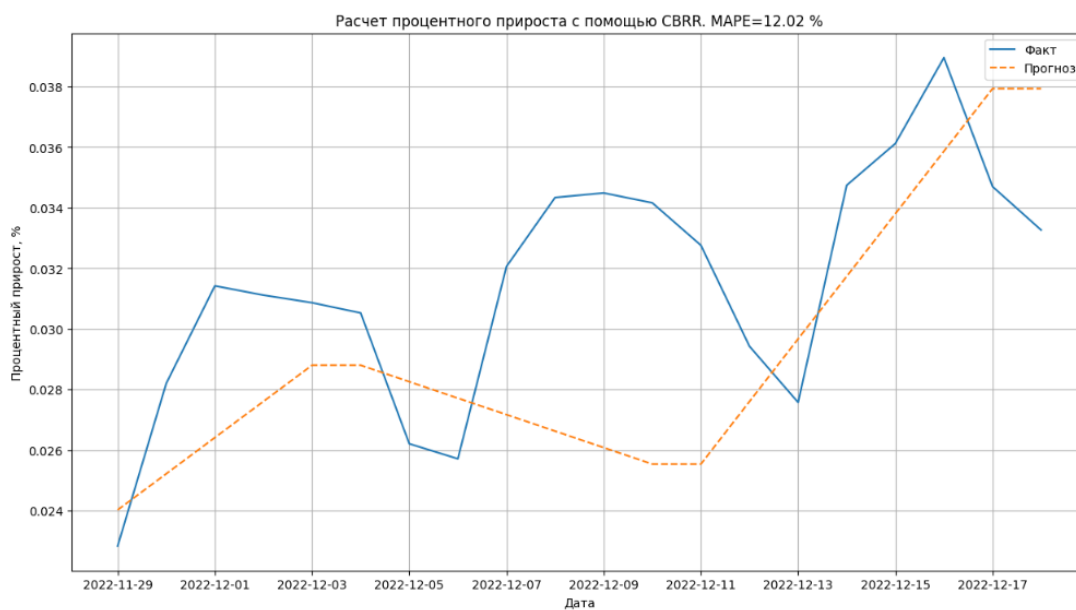


Рисунок 37. Прогноз динамики процентного прироста в период с 29.11.2022 по 17.12.2022.

Получив значения $\tilde{r}(T_{m-1} + k)$, построим прогноз для общего количества выявленных случаев.

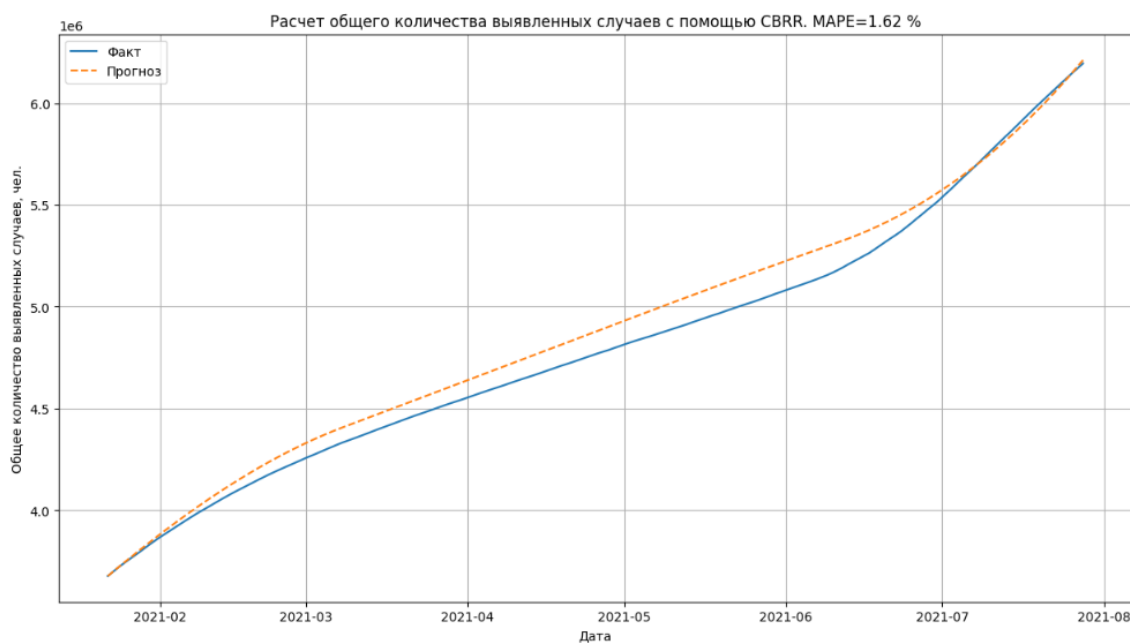


Рисунок 38. Расчет общего количества выявленных случаев в период с 22.01.2021 по 27.07.2021

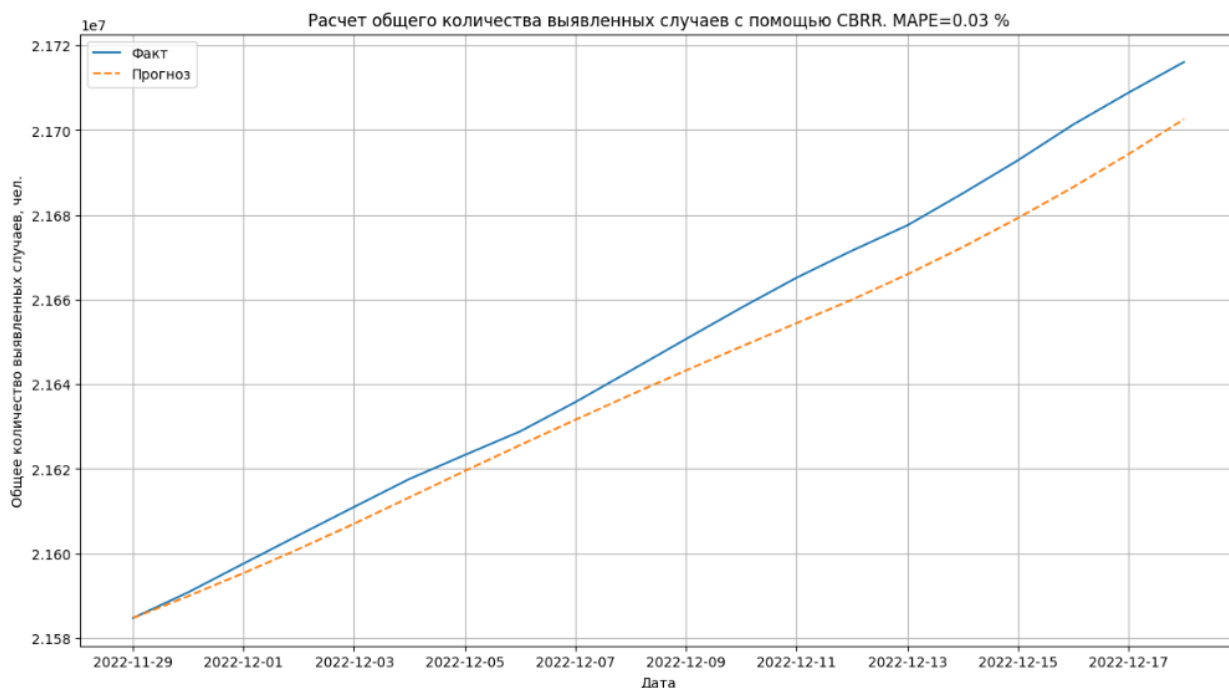


Рисунок 39. Расчет общего количества выявленных случаев в период с 29.11.2022 по 17.12.2022

Используя метод прецедентов на основе процентного прироста, получен прогноз динамики общего количества выявленных случаев в России. В первом промежутке средняя ошибка составила 1.62%, во втором 0.03%. Данный прогноз будет использован далее для оценки количества зараженных людей.

4.2 БАЛАНСОВАЯ МОДЕЛЬ CIR

Балансовая модель CIR была предложена в 2021 году в работе [1]. Как уже было сказано ранее во введении, особенность данной модели заключается в том, что она способна учитывать историю развития пандемии, используя принцип динамического баланса. Графическая интерпретация данного принципа представлена на рис.40.

Рассматривается промежуток $[t_0; T]$, в котором количество выздоровевших и умерших в момент времени T больше, чем общее количество выявленных случаев от t_0 ($R(T) > C(t_0) > 1$).

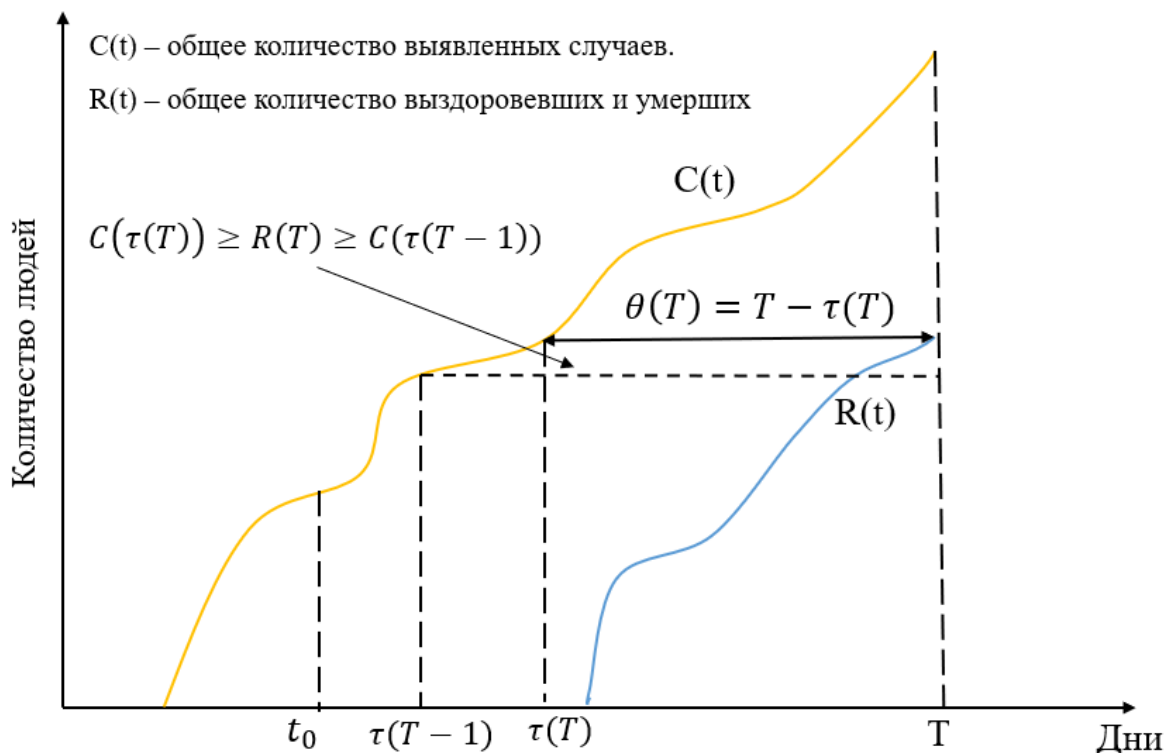


Рисунок 40. Иллюстрация принципа динамического баланса

В этом промежутке решается следующая задача: находим первый момент времени, при котором $C(t) \geq R(T)$:

$$\begin{cases} \min_{t_0 \leq t \leq T} t \\ C(t) \geq R(T) \end{cases}$$

Так как функции $C(t), R(t)$ неубывающие, то такое решение существует. Обозначим этот момент времени через $\tau(T)$.

Согласно принципу динамического баланса, при сделанных условиях, значение $R(T)$ лежит в промежутке между двумя точками:

$$C(\tau(T)) \geq R(T) \geq C(\tau(T) - 1)$$

Величина $\theta(T) = T - \tau(T)$ называется характеристика динамического баланса.

В принятых условиях функция $R(T)$ представима в виде формулы для внутренней точки отрезка соединяющего две точки:

$$R(T) = \lambda_T C(\tau(T) - 1) + (1 - \lambda_T) C(\tau(T)), \quad \lambda_T \in [0; 1]$$

Важно подчеркнуть, что общее количество подтвержденных случаев $C(t)$ складывается из суммы числа инфицированных $I(t)$ и количества выздоровевших и умерших $R(t)$:

$$C(t) = I(t) + R(t)$$

Воспользовавшись представлением $C(t)$ на основе процентного прироста, запишем систему дискретных уравнений, которая описывает модель CIR:

$$\begin{cases} C(t) = \left(1 + \frac{r(t)}{100}\right) \times C(t - 1) \\ I(t) = C(t) - R(t) \\ R(T) = \lambda_T C(\tau(T) - 1) + (1 - \lambda_T) C(\tau(T)) \end{cases}$$

Таким образом, чтобы предсказать количество болеющих людей (именно эта группа оказывает наибольшую нагрузку на систему здравоохранения) необходимо оценить общее количество подтвержденных случаев $C(t)$ и количество выздоровевших и умерших $R(t)$. Динамика $C(t)$ была предсказана в прошлом пункте с помощью метода прецедентов. Для предсказания $R(t)$ воспользуемся полученным результатом прогнозирования ХДБ из главы 3, найдем момент времени $\tau(T)$ и возьмем нижнюю границу оценки $C(\tau(T) - 1)$ из принципа динамического баланса.

На рис.41 показан краткосрочный прогноз количества инфицированных людей. На этом промежутке удалось предсказать ХДБ с малой ошибкой (МАРЕ=1.86%). Прогноз динамики ХДБ позволил снизить среднюю абсолютную ошибку предсказания количества болеющих людей на 2%.

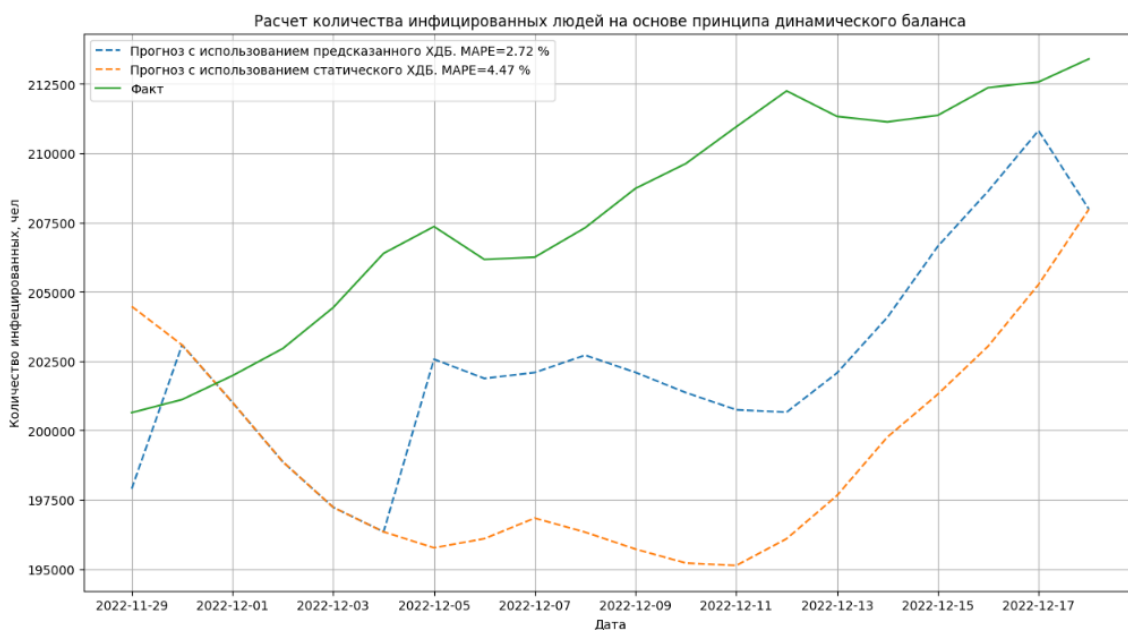


Рисунок 41. Прогноз количества инфицированных людей за период с 29.11.2022 по 17.12.2022

Посмотрим какой будет прогноз при долгосрочном предсказании.

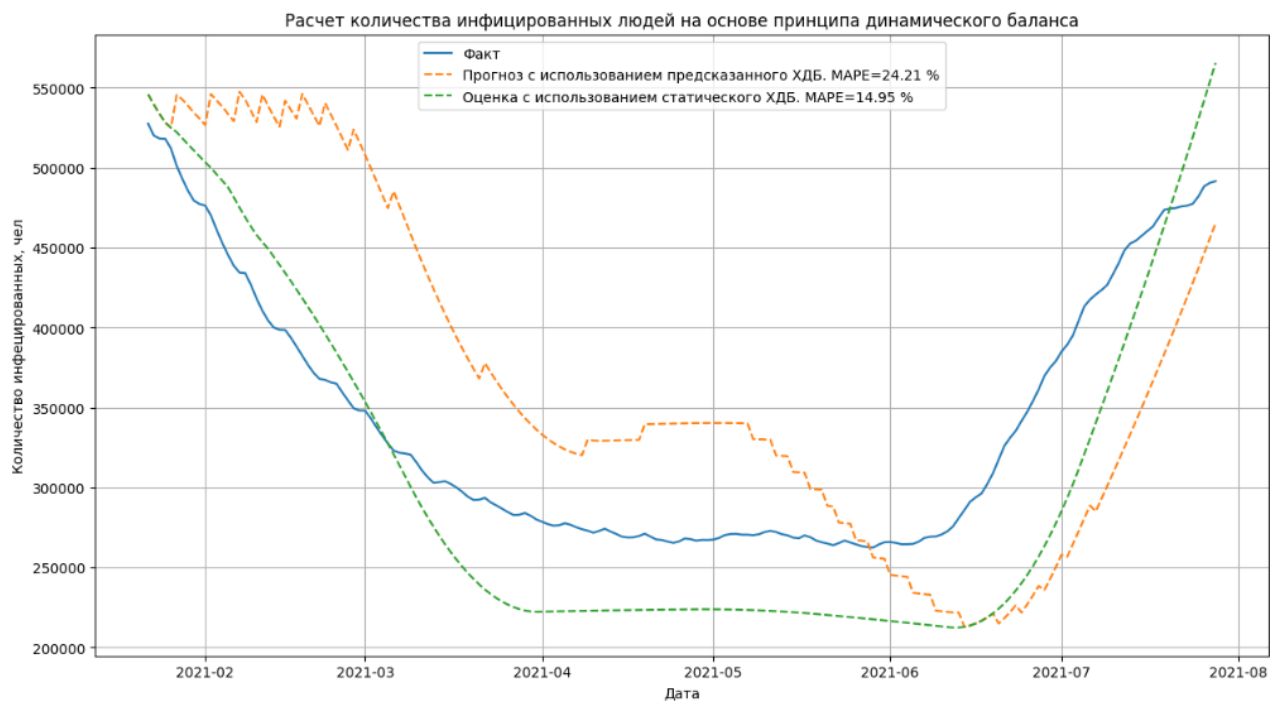


Рисунок 42. Прогноз количества инфицированных людей за период с 22.01.2021 по 27.07.2021

На втором промежутке ошибка увеличилась на 10%, если использовать предсказанную динамику ХДБ. Качества прогноза модели DA RNN на рассматриваемом промежутке оказалось недостаточно, чтобы добиться улучшения (напомню, MAPE=14.12% при моделировании значений ХДБ)

Рассмотрим отрезок времени, в котором происходит спад динамики ХДБ. По ранее описанной методике оценим количество болеющих людей.

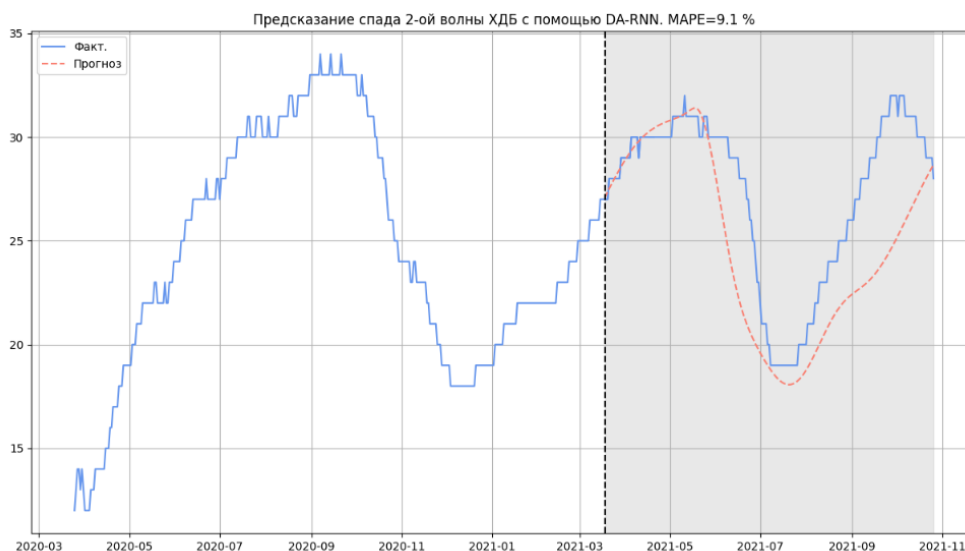


Рисунок 43. Предсказание спада 2-й волны ХДБ.

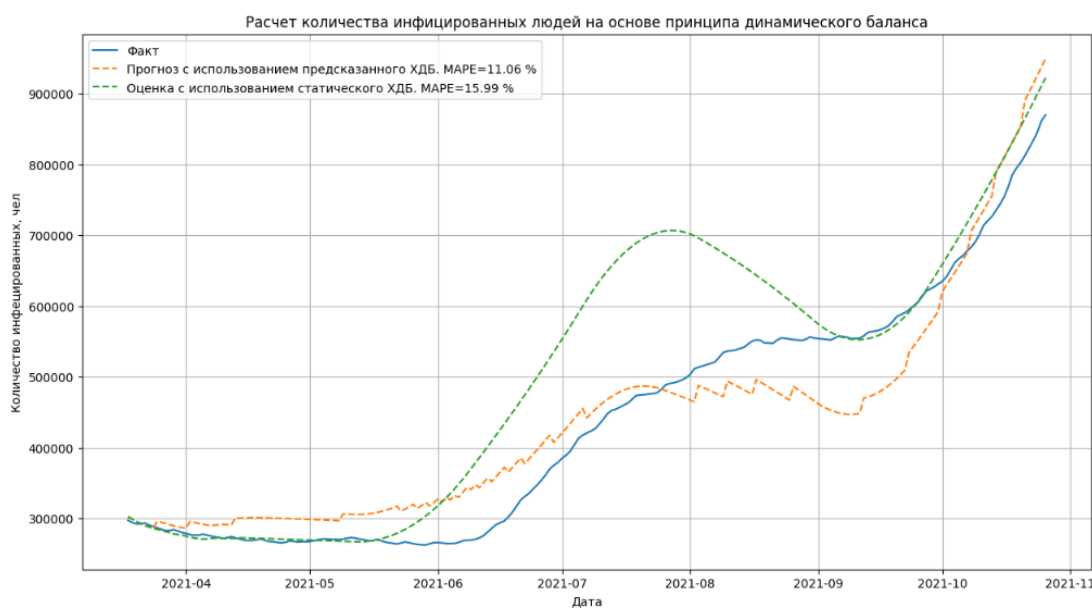


Рисунок 44. Расчет количества инфицированных людей на основе принципа динамического баланса в период с 18.03.2021 по 26.10.2021.

В отрезке с 18.03.2021 по 26.10.2021 средняя абсолютная ошибка предсказания ХДБ составила 9.1%. Такой точности оказалось достаточно, чтобы улучшить долгосрочное предсказание на 4%.

Подводя итог данной главы, можно сделать вывод, что на краткосрочном промежутке моделирования удалось повысить качество предсказания модели SIR, используя прогнозные значения ХДБ. Но стоит учесть, что для улучшения результата необходимо добиться достаточно высокой точности в прогнозировании ХДБ.

ЗАКЛЮЧЕНИЕ

Прогнозирование динамики параметров распространения пандемии COVID-19 является сложной и нестандартной задачей. Трудности в исследовании возникают из-за нехватки информации о поведении вируса в разных регионах. В своей работе я занимался прогнозированием такого параметра как характеристика динамического баланса. В ходе исследования были решены следующие задачи:

1. Проанализированы статистические методы прогнозирования временных рядов. Лучший результат у модели линейной регрессии с несколькими предикторами (MAPE=18.03%)
2. Проведен обзор наиболее популярных структур рекуррентных нейронных сетей. С их помощью удалось снизить среднюю ошибку (лучшая метрика MAPE=12.03% у модели Vi-LSTM), но не получилось поймать тренд динамики.
3. Для улучшения результата была реализована модель DA RNN основанная на двухэтапном механизме внимания. Модель предсказала значение ХДБ с наименьшим отклонением (MAPE=6.23%), а также хорошо улавливает тренд динамики.
4. Предсказанные значения ХДБ позволили улучшить точность краткосрочного прогноза количества инфицированных людей, основываясь на модели CIR. Ошибка при использовании стационарного значения ХДБ составила 4.47%, а при использовании прогнозных значений 2.72%

Предложенная модель прогнозирования не лишена недостатков:

1. Основной ее минус, как и у всех нейронных сетей, заключается в невозможности интерпретации результата. Из-за этого оценить поведение модели на разных данных не представляется возможным.
2. Не удалось добиться качественного предсказания в долгосрочном прогнозировании. Однако, краткосрочное прогнозирование также

является ограничением модели CIR (авторы указали это в работе [1]). Поэтому полученной точности достаточно для улучшения оценки количества активно болеющих

В качестве дальнейшего улучшения результата прогнозирования можно добавить сверточный слой в модели DA RNN. Также открытыми остаются вопросы об оптимальном выборе значений процентного прироста при предсказании общего числа выявленных случаев и вычислении коэффициента λ_T при оценке количества выздоровевших и умерших в принципе динамического баланса.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Захаров В. В., Балыкина Ю. Е. Балансовая модель эпидемии COVID-19 на основе процентного прироста //Информатика и автоматизация. – 2021. – Т. 20. – №. 5. – С. 1034-1064.
2. Захаров В. В., Балыкина Ю. Е. Прогнозирование динамики эпидемии коронавируса (COVID-19) на основе метода прецедентов //Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления. – 2020. – №. 3. – С. 249-259.
3. AlQadi H., Bani-Yaghoub M. Incorporating global dynamics to improve the accuracy of disease models: Example of a COVID-19 SIR model //Plos one. – 2022. – Т. 17. – №. 4. – С. e0265815.
4. Annas S. et al. Stability analysis and numerical simulation of SEIR model for pandemic COVID-19 spread in Indonesia //Chaos, solitons & fractals. – 2020. – Т. 139. – С. 110072.
5. Arora P., Kumar H., Panigrahi B. K. Prediction and analysis of COVID-19 positive cases using deep learning models: A descriptive case study of India //Chaos, Solitons & Fractals. – 2020. – Т. 139. – С. 110017.
6. Bahdanau D., Cho K., Bengio Y. Neural machine translation by jointly learning to align and translate //arXiv preprint arXiv:1409.0473. – 2014.
7. Bartholomew D. J. Time Series Analysis Forecasting and Control. – 1971.
8. Bontempi G., Ben Taieb S., Le Borgne Y. A. Machine learning strategies for time series forecasting //Business Intelligence: Second European Summer School, eBISS 2012, Brussels, Belgium, July 15-21, 2012, Tutorial Lectures 2. – 2013. – С. 62-77.
9. Borghi P. H., Zakordonets O., Teixeira J. P. A COVID-19 time series forecasting model based on MLP ANN //Procedia Computer Science. – 2021. – Т. 181. – С. 940-947.
10. Chin A. W. H. et al. Stability of SARS-CoV-2 in different environmental conditions //The Lancet Microbe. – 2020. – Т. 1. – №. 1. – С. e10.

11. Crone S. F., Hibon M., Nikolopoulos K. Advances in forecasting with neural networks? Empirical evidence from the NN3 competition on time series prediction //International Journal of forecasting. – 2011. – T. 27. – №. 3. – C. 635-660.
12. Djakaria I., Saleh S. E. Covid-19 forecast using Holt-Winters exponential smoothing //Journal of Physics: Conference Series. – IOP Publishing, 2021. – T. 1882. – №. 1. – C. 012033.
13. Dutta S., Bandyopadhyay S. K. Machine learning approach for confirmation of COVID-19 cases: positive, negative, death and release [Preprint]. – 2020.
14. Fang Z. et al. Application of a data-driven XGBoost model for the prediction of COVID-19 in the USA: a time-series study //BMJ open. – 2022. – T. 12. – №. 7. – C. e056685.
15. Guhathakurata S. et al. A novel approach to predict COVID-19 using support vector machine //Data Science for COVID-19. – Academic Press, 2021. – C. 351-364.
16. Hernandez-Matamoros A. et al. Forecasting of COVID19 per regions using ARIMA models and polynomial functions //Applied soft computing. – 2020. – T. 96. – C. 106610.
17. Hochreiter S., Schmidhuber J. Long short-term memory //Neural computation. – 1997. – T. 9. – №. 8. – C. 1735-1780.
18. Ilu S. Y., Rajesh P., Mohammed H. Prediction of COVID-19 using long short-term memory by integrating principal component analysis and clustering techniques //Informatics in Medicine Unlocked. – 2022. – T. 31. – C. 100990.
19. Karpathy A., Fei-Fei L. Deep visual-semantic alignments for generating image descriptions //Proceedings of the IEEE conference on computer vision and pattern recognition. – 2015. – C. 3128-3137.
20. Luo J. et al. Time series prediction of COVID-19 transmission in America using LSTM and XGBoost algorithms //Results in Physics. – 2021. – T. 27. – C. 104462.

21. Panda M. Application of ARIMA and Holt-Winters forecasting model to predict the spreading of COVID-19 for India and its states //medRxiv. – 2020. – C. 2020.07. 14.20153908.
22. Pandey G. et al. SEIR and Regression Model based COVID-19 outbreak predictions in India //arXiv preprint arXiv:2004.00958. – 2020.
23. Parmezan A. R. S., Souza V. M. A., Batista G. E. Evaluation of statistical and machine learning models for time series prediction: Identifying the state-of-the-art and the best conditions for the use of each model //Information sciences. – 2019. – T. 484. – C. 302-337.
24. Petchko K. How to write about economics and public policy. – Academic Press, 2018.
25. Qin Y. et al. A dual-stage attention-based recurrent neural network for time series prediction //arXiv preprint arXiv:1704.02971. – 2017.
26. Shoko C., Sigauke C. Short-term forecasting of COVID-19 using support vector regression: An application using Zimbabwean data //American Journal of Infection Control. – 2023.
27. Wang Y. et al. Prediction and analysis of COVID-19 daily new cases and cumulative cases: times series forecasting and machine learning models //BMC Infectious Diseases. – 2022. – T. 22. – №. 1. – C. 1-12.
28. Werbos P. J. Generalization of backpropagation with application to a recurrent gas market model //Neural networks. – 1988. – T. 1. – №. 4. – C. 339-356.
29. Zakharov V. et al. CBRR model for predicting the dynamics of the COVID-19 epidemic in real time //Mathematics. – 2020. – T. 8. – №. 10. – C. 1727.