Министерство науки и высшего образования 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

Вероятностное моделирование развития пандемии средствами интеллектуального анализа данных

по направлению подготовки 01.04.03 – «Механика и математическое моделирование»

профиль 01.04.03_04 – «Механика и цифровое производство»

Выполнила
студентка гр. 5040103/10301 М.А. Курдина

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

«__»_____2023 г.

ЗАДАНИЕ

на выполнение выпускной квалификационной работы

студенту Курдиной Марии Алексеевне группы 5040103/10301

1. Тема работы: Вероятностное моделирование развития пандемии средствами интеллектуального анализа данных

2. Срок сдачи студентом законченной работы: 26.05.2023

3. Исходные данные по работе: научные статьи по прогнозированию развития эпидемий и пандемий, статистика по заболеваемости коронавирусом Covid-19 за период с 09.03.2020 г. до 05.03.2023 г. и статистика по вакцинации за период с 14.02.2021 г. до 05.03.2023 г.

4. Содержание работы (перечень подлежащих разработке вопросов): анализ существующих методов прогнозирования временных рядов; прогнозирование количества людей, зараженных Covid-19, на основе исторических данных по заболеваемости; исследование влияния вакцинации на заболеваемость; прогнозирование развития пандемии в зависимости от количества вакцинированных людей с использованием различных алгоритмов машинного обучения и статистических методов; вероятностное прогнозирование развития пандемии в зависимости от количества вакцинированных людей с использованием различных алгоритмов машинного обучения и статистических методов.

5. Перечень графического материала (с указанием обязательных чертежей): графики временных рядов с количеством заражений и количеством вакцинаций

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 _____ М.А. Курдина
(подпись) инициалы, фамилия

РЕФЕРАТ

На 73 с., 28 рисунков, 7 таблиц

КЛЮЧЕВЫЕ СЛОВА: ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ, АНАЛИЗ ДАННЫХ, МАШИННОЕ ОБУЧЕНИЕ, СТАТИСТИЧЕСКИЕ МОДЕЛИ, ПРОГНОЗИРОВАНИЕ ВРЕМЕННЫХ РЯДОВ, ВЕРОЯТНОСТНОЕ ПРОГНОЗИРОВАНИЕ, ПАНДЕМИЯ, COVID-19, PYTHON.

Тема выпускной квалификационной работы: «Вероятностное моделирование развития пандемии средствами интеллектуального анализа данных».

Данная работа посвящена исследованию возможности применения вероятностного моделирования для прогнозирования развития пандемии Covid-19 и созданию инструмента для предварительной обработки данных и прогнозирования временных рядов на языке Python. Для повышения точности прогнозирования анализируется возможность применения статистики по вакцинации в качестве дополнительного предиктора в прогнозной модели с помощью оценки влияния вакцинации на заболеваемость. Также в данной работе производится сравнение двух статистических моделей и трех моделей машинного обучения для предсказания заболеваемости коронавирусом. Рассматриваются методы экспоненциального сглаживания и ARIMA в качестве статистических подходов к прогнозированию и алгоритмы машинного обучения: k-ближайших соседей, случайный лес, градиентный бустинг. Выбор оптимального алгоритма производится на основе ошибки MAPE, полученной при прогнозе на тестовой выборке, наиболее точного соответствия характеру исходного временного ряда и скорости работы. Результаты исследования показывают, что оптимальным алгоритмом для прогнозирования является градиентный бустинг. Результаты данной работы актуальны, т.к. могут быть полезны для медицинских организаций, государственных структур при принятии мер для предотвращения распространения будущих вспышек вирусных заболеваний.

ABSTRACT

73 pages, 28 pictures,7 tables

KEYWORDS: ARTIFICIAL INTELLIGENCE, DATA ANALYSIS, MACHINE LEARNING, STATISTICAL MODELS, TIME SERIES FORECASTING, PROBABILISTIC FORECASTING, PANDEMIC, COVID-19, PYTHON.

The subject of the graduate qualification work: «Probabilistic forecasting of the evolution of a pandemic based on intelligent data analysis»

This paper is devoted to the study of the possibility of using probabilistic modeling to predict the evolution of the Covid-19 pandemic and creating a tool for data pre-processing and time-series forecasting in Python. To improve forecasting accuracy, the possibility of using vaccination statistics as an exogenous variable of the forecast model is analyzed by estimating the effect of vaccination on incidence. In this paper, two statistical models and three machine learning models for predicting the incidence of coronavirus are compared. Exponential smoothing and ARIMA methods are considered as statistical approaches for prediction and k-nearest neighbors, random forest, and gradient binning algorithms as machine learning methods. The choice of the optimal algorithm is made based on the MAPE obtained from the prediction on the test sample, as well as the best fit to the pattern of the original time series and the speed of the performance. The results of the study show that the optimal algorithm for prediction is the gradient boosting model. The results of this work are relevant, because it can be useful for medical organizations and the government in taking measures to prevent the spread of future epidemics of virus diseases.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Новая коронавирусная инфекция Covid-19 распространилась по всему миру и оказала огромное влияние на жизнь людей и национальное развитие во всем мире. Вирус Covid-19 создал социальные, политические и экономические вызовы для человечества. Все страны мира сделали все возможное, чтобы контролировать распространение вируса, как с точки зрения человеческих, так и финансовых ресурсов. Но, к сожалению, этого не хватило для полного устранения угрозы. Именно поэтому для своевременного реагирования исследователи из разных стран вынуждены искать новые способы борьбы с пандемией.

Прогнозирование развития пандемии, в том числе Covid -19, является важной и актуальной задачей, так как может быть использоваться в принятии социальных, экономических и политических решений. Прогноз количества заболевших людей поможет политикам в конкретном регионе оценить их текущий потенциал здравоохранения и решить, какие меры необходимо принять для сдерживания и контроля распространения пандемий. Чтобы дать более точный прогноз развития пандемии, необходимо проводить исследования и анализировать данные, которые могут влиять на ее развитие. Например, исследования могут быть направлены на выявление особенностей распространения вируса в зависимости от климатических условий, уровня вакцинации, наличия иммунитета и многих других факторов. С момента появления Covid-19 в конце 2019 года исследователи изучали характер и скорость заражения новой коронавирусной инфекцией с использованием различных методологий математического моделирования, популярных в эпидемиологических исследованиях.

Искусственный интеллект (ИИ) – многообещающая технология для решения задач здравоохранения, которая может облегчить процессы, связанные с анализом больших объемов данных. Это позволит более точно определять тенденции и прогнозировать развитие ситуации в нужных областях. ИИ применяется в отслеживании и прогнозировании эпидемии или пандемии, а также

разработке методов профилактики и лечения вируса. По мере поступления новых данных искусственный интеллект обучается, точность решений повышается.

ИИ базируется на системах, способных обрабатывать данные и на их основе принимать решения. Они призваны организовать работу с большими объёмами информации автоматически, без вмешательства человека. Ветви искусственного интеллекта представляют собой алгоритмы машинного обучения. Интеллектуальная система на основе машинного обучения позволяет провести поиск закономерностей и составлять прогнозы на основе полученной модели.

В связи с возрастанием интереса к использованию интеллектуальных методов для прогнозирования развития пандемий возникла необходимость в использовании сложных моделей и алгоритмов, которые могут обеспечить более точные прогнозы. В данной работе исследуется возможность применения вероятностного моделирования, основанного на интеллектуальном анализе данных, для прогнозирования распространения коронавируса. Также анализируется влияние вакцинации на распространение вируса и на точность прогноза. Для этой цели используются модели, основанные на машинном обучении и статистическом анализе данных.

Полученные в результате прогнозные модели развития пандемий, основанные на алгоритмах машинного обучения, могут оказаться очень полезным для принятия защитных мер и разработки плана действий во время таких пандемий, как Covid-19.

ГЛАВА 1. АНАЛИЗ СУЩЕСТВУЮЩИХ МЕТОДОВ ДЛЯ ПРОГНОЗИРОВАНИЯ ВРЕМЕННЫХ РЯДОВ

Временные ряды используются во многих областях, включая финансы, маркетинг, медицину и другие. Они представляют собой последовательность значений, которые изменяются во времени. Прогнозирование временных рядов позволяет предсказывать будущие значения на основе прошлых данных. Для этого существуют различные методы анализа временных рядов.

В эпидемиологии принято использовать методы моделирования для определения скорости и тенденции распространения инфекционных заболеваний. Существует множество методов моделирования, которые исследователи применяли для прогнозирования эпидемий. Моделирование инфекционных заболеваний классифицируется на 3 типа: статистическое методы эпидемического мониторинга, математические и/или механические модели в пространстве состояний, а также эмпирические модели и/или модели машинного обучения.

В данной главе будет проведен анализ существующих методов прогнозирования временных рядов на основе научных работ. В ходе исследования будут проанализированы различные методы, включая статистические алгоритмы и модели машинного обучения. Результаты анализа помогут выбрать наиболее подходящий метод для прогнозирования временных рядов.

1.1. Обзор существующих статистических методов для прогнозирования временных рядов

Существует множество различных статистических методов для прогнозирования временных рядов, включая линейные модели, модели авторегрессии, модели авторегрессии с экспоненциальным сглаживанием, модели авторегрессии с двумя параметрами, модели авторегрессии с компонентами сезонности и многие другие. Каждый из этих методов имеет свои преимущества и недостатки, поэтому важно понимать их применение и особенности.

В данном разделе будет рассмотрено два статистических метода для прогнозирования временных рядов. Также будут проанализированы статьи, в которых используются выбранные методы для прогнозирования заболеваемости COVID-19.

1.1.1. ARIMA

Одним из наиболее популярных методов прогнозирования временных рядов является модель ARIMA *(autoregressive integrated moving average*). Это метод авторегрессии временных рядов, который вычисляет будущие краткосрочные прогнозы на основе анализа исторических данных временного ряда. Модель ARIMA была разработана Боксом и Дженкинсом в 1970-х годах для математического описания изменений временного ряда [1]. В модели ARIMA используется авторегрессионный компонент (AR) для описания зависимости между текущими и предыдущими значениями ряда, интегрированный компонент (I) для описания изменений тренда и скользящее среднее (MA) для описания изменений сезонности. ARIMA (p, q, d) обозначает модель с авторегрессионными задержками p, задержками скользящего среднего q и интегрированным компонентом порядка d. Эта модель является расширением модели ARMA для нестационарных временных рядов, которые можно привести к стационарным, путём взятия производной разного порядка от исходного временного ряда. Таким образом, модель ARIMA позволяет более точно прогнозировать будущие значения временных рядов и использовать эти прогнозы для принятия обоснованных решений в различных областях деятельности.

Множество исследований подтверждают, что тщательный и точный выбор модели ARIMA может быть применен к временным рядам с одной переменной, любым паттерном и автокорреляцией между последовательными значениями ряда во времени для более точного прогнозирования будущих значений.

В статье [19] рассматривается применение модели ARIMA для прогнозирования производства сахарного тростника в Индии. Основная причина выбора модели ARIMA в этом исследовании для прогнозирования заключается в том, что эта модель предполагает и учитывает ненулевую автокорреляцию между последовательными значениями данных временного ряда. Порядок лучшей модели ARIMA оказался (2,1,0). Кроме того, были предприняты меры для повышения точности прогнозирования будущего производства сахарного тростника на период до пяти лет путем подгонки модели ARIMA(2,1,0) к данным временного ряда. Исследование также статистически проверило и подтвердило, что ошибки прогноза в аппроксимированных временных рядах ARIMA не коррелировали, и остатки нормально распределены с нулевым средним значением и постоянной дисперсией. Следовательно, выбранный вариант модели ARIMA обеспечивает адекватную прогностическую модель для производства сахарного тростника в Индии.

Авторы статьи [5] также использовали модель ARIMA для прогнозирования будущей стоимости биткойна путем анализа временных рядов цен за трехлетний период времени. Исследование показало, что с одной стороны, что эта простая схема эффективна в подпериодах, в которых поведение временного ряда почти не меняется, особенно когда она используется для краткосрочного прогнозирования, например, на 1 день. С другой стороны, при обучении модели ARIMA на трехлетнем периоде, в течение которого цена биткойна демонстрировала различное поведение, или когда модель используется для долгосрочного прогноза, наблюдаются большие ошибки предсказания. В частности, модель ARIMA не может уловить резкие колебания цены. Затем она требует извлечения и использования дополнительных функций вместе с ценой для более точного прогнозирования цены.

ARIMA также использовалась для прогнозирования нескольких вспышек заболеваний, таких как энтеровирусный везикулярный стоматит (англ. HFMD) в Китае [23], гепатит-B [39], а также вируса COVID-19 [8].

Модель ARIMA представляет собой мощный инструмент для прогнозирования временных рядов. Однако для достижения максимальной точности прогнозирования необходимо правильно подобрать параметры модели и проверить ее качество.

Хотя, как и любые другие прогностические модели в прогнозировании, ARIMA также имеет ограничения по точности прогнозов, тем не менее, она широко используется для прогнозирования будущих последовательных значений во временном ряду.

1.1.1 Экспоненциальное сглаживание

Еще одним популярным методом прогнозирования временных рядов является экспоненциальное сглаживание. Формулировка методов прогнозирования экспоненциального сглаживания возникла в 1950-х годах на основе оригинальной работы Брауна [2] и Хольта [16], которые работали над созданием моделей прогнозирования.

Экспоненциальное сглаживание является интуитивно понятным методом прогнозирования, при котором наблюдаемые временные ряды взвешиваются неравномерно. Недавние наблюдения имеют больший вес, чем более старые наблюдения. Неравное взвешивание достигается за счет использования одного или нескольких параметров сглаживания, которые определяют, какой вес придается каждому наблюдению. Обычно для этого используется взвешенное скользящее среднее.

Простое экспоненциальное сглаживание подходит для ряда, который случайным образом перемещается выше и ниже постоянного среднего значения, т.е. стационарный ряд. Такой временной ряд не имеет тренда и сезонных закономерностей. Обычно используется в краткосрочном прогнозировании.

Двойное и тройное экспоненциальное сглаживание представляют собой расширение экспоненциального сглаживания. Эти подходы широко

используются для прогнозирования временных рядов, содержащих изменяющийся тренд и сезонность.

Популярность экспоненциального сглаживания объясняется его простотой, вычислительной эффективностью, легкостью настройки реакции на изменения в прогнозируемом процессе и его приемлемой точностью [4].

В [33] целью исследования является наблюдение за прогнозированием продаж продукта. Для поставленной цели используются методы: простое экспоненциальное сглаживание и двойное экспоненциальное сглаживание. Результат показывает, что средняя абсолютная ошибка в процентах, MAPE, экспоненциального сглаживания составляет 20%, а MAPE двойного экспоненциального сглаживания – около 24%. Использование метода простого экспоненциального сглаживания имеет меньшую ошибку. Исследование показывает, что в этом случае рекомендуется использовать прогнозирование с однократным экспоненциальным сглаживанием. Таким образом, данные, обработанные этими алгоритмами, будут полезным результатом для определения запасов продукции в будущем.

В статье [29] применяется простое экспоненциальное сглаживание для прогнозирования первичного производства электроэнергии в Словакии. Для оценки точности модели были использованы три метрики, такие как средняя абсолютная ошибка (MAE), средняя абсолютная ошибка в процентах (MAPE) и среднеквадратичная ошибка (RMSE). На основе результатов метрик выбирается наиболее точный прогноз на год вперед.

В целом экспоненциальное сглаживание рассматривается как недорогой метод, который дает хорошие результаты прогноза. Кроме того, требования к хранению данных и вычислительным ресурсам минимальны, что делает экспоненциальное сглаживание пригодным для промышленных приложений, работающих в реальном времени.

1.1.2 Использование статистических методов для прогнозирования заболеваемости Covid-19

Статистические методы являются широко используемым инструментом не только для прогнозирования в сфере маркетинга, финансов, производства, но и в медицине. В том числе используются для прогнозирования развития эпидемий или пандемий.

Модель ARIMA не активно использовалась исследователями для прогнозирования COVID-19 из-за утверждения, что она не подходит для использования в сложных и динамичных условиях. Исследователи в [6] провели анализ, насколько точными являются прогнозы модели ARIMA. В данной статье подтверждается хорошая точность моделей ARIMA для прогнозирования развития пандемии Covid-19 в течение относительно длительного периода времени на примере Кувейта.

Помимо этого, исследователи во многих странах использовали модель ARIMA для прогнозирования распространения пандемии COVID-19. В [8] произведено сравнение нескольких моделей ARIMA с разными параметрами для оценки распространения COVID-19 в Испании, Франции и Италии. Авторы статьи [31] также вносят свой вклад в текущую катастрофу, используя модель ARIMA для предсказания увеличения числа случаев COVID-19 в течение следующих 15 дней в Индии. Эти исследования показывают, что модели ARIMA подходят для прогнозирования распространенности COVID-19 в будущем.

Модель ARIMA была также используется при прогнозировании COVID-19 в период с 19 февраля до 29 апреля 2020 года в Иране и сравнивается с искусственными нейронными сетями [27]. Сравнение результатов работы моделей в исследовании показало, что прогноз ARIMA был более точным, чем прогноз на основе нейронных сетей.

Экспоненциальное сглаживание также используется для предсказания развития пандемии. Например, в [11] применяется модель тройного экспоненциального сглаживания Хольта-Винтерса для прогнозирования развития пандемии в определенный период с 10 апреля по 13 октября 2020 года в

Индонезии. Результаты показывают, что с помощью экспоненциального сглаживания Хольта-Винтерса удалось достигнуть точного прогноза с наименьшим значением ошибки MAPE, равной 6%.

Таким образом, статистические методы прогнозирования имеют большое значение для предотвращения распространения заболевания Covid-19. Они позволяют предсказывать динамику заболеваемости в различных регионах и принимать наиболее эффективные меры по предотвращению распространения инфекции.

1.2. Обзор существующих методов машинного обучения для прогнозирования временных рядов

Прогнозирование временных рядов традиционно выполнялось с использованием статистических методов, таких как модели ARIMA или экспоненциальное сглаживание. Однако в последние десятилетия стали использоваться методы интеллектуального анализа данных для прогнозирования временных рядов.

Искусственный интеллект, машинное обучение и наука о данных связаны друг с другом. Машинное обучение можно рассматривать как подобласть или один из инструментов искусственного интеллекта, предоставляя машинам возможность обучения.

Машинное обучение признано для решения многих задач в реальном времени, включая обработку изображений, медицинскую диагностику, финансовый анализ, прогнозирование и т.д. Различные прогнозы, в том числе погода, национальный фондовый рынок и многие другие, используют алгоритмы машинного обучения для прогнозирования будущего, чтобы были предприняты необходимые действия.

Алгоритмам машинного обучения требуется достаточное количество данных для лучшего прогнозирования. По мере увеличения размера обучающего набора данных производительность модели увеличивается.

В данном разделе будут рассмотрены некоторые алгоритмы для прогнозирования временных рядов, в том числе для прогнозирования заболеваемости вирусом COVID-19.

1.2.1. Алгоритм k-ближайших соседей

Метод k-ближайших соседей (k-Nearest Neighbors или kNN) – популярный алгоритм машинного обучения для решения задач классификации и регрессии.

В этой работе [35] предлагаемый метод для прогнозирования потреблении энергии основан на методе ближайших соседей. Этот выбор обусловлен хорошими результатами, полученными при применении к наборам данных небольшого или среднего размера. Также алгоритм был проверен на реальных наборах больших данных.

В статье [26] также используется регрессия KNN для прогнозирования одномерных временных рядов. Авторы показали, как регрессия KNN может применяться в контексте прогнозирования временных рядов.

Выбор числа соседей и признаков представляет собой сложную задачу. В [34] представлены две методологии прогнозирования временных рядов: классическая настройка параметров во взвешенных ближайших соседях и быстрая настройка параметров во взвешенных ближайших соседях. Полученные модели сравниваются с некоторыми классическими подходами: ARIMA и моделью Хольта-Винтерса. Также приводится оценка эффективности и точности модели. Проанализированы реальные примеры данных о розничных продажах и продажах общественного питания в США и производстве молока в Великобритании, чтобы продемонстрировать применение и эффективность предложенных подходов.

Таким образом, несмотря на свою простоту, метод k-ближайших соседей успешно применяется в прогнозировании временных рядов, обеспечивая конкурентоспособные результаты. Однако эти методы нельзя применять, когда

необходимо прогнозировать большие временные ряды из-за высокой вычислительной стоимости алгоритма.

1.2.2. Ансамблевые модели

Ансамблевые модели в машинном обучении – это модели, которые используют несколько алгоритмов машинного обучения для построения более точной модели. Они могут использоваться для предсказания или классификации данных. Наиболее распространенные ансамблевые модели включают бэггинг, бустинг и стекинг.

1.2.2.1 Случайный лес

Случайный лес – это алгоритм машинного обучения, который использует комбинацию «деревьев решений» для предсказания вероятности или значения выходной переменной. Алгоритм строится на основе концепции бэггинга. Само по себе решающее дерево предоставляет крайне невысокое качество предсказания, но из-за большого количества деревьев результат значительно улучшается. Стоит отметить, что для задачи регрессии решение определяется усреднением по всем решающим деревьям.

Случайный лес при работе с временными рядами не учитывает зависящую от времени структуру, предполагая, что наблюдения независимы, поэтому требуется адаптация модели к прогнозированию временных рядов.

Авторы статьи [15] предлагают несколько вариантов случайных лесов, разработанных для прогнозирования временных рядов. Идея состоит в том, чтобы заменить стандартную бутстрэп-выборку зависимой бутстрэп-выборкой для временных рядов на этапе построения дерева, чтобы учесть временную зависимость. Затем производится два эксперимента по прогнозированию нагрузки на электроэнергию.

Помимо этого, алгоритм случайного леса успешно применяется прогнозирование временных рядов во многих исследованиях [12, 21, 13, 28].

1.2.2.2 Градиентный бустинг

Алгоритм градиентного бустинга представляет собой алгоритм итеративного дерева решений [14]. Он также может автоматически обрабатывать отсутствующие значения и ненормальные значения.

Алгоритм XGBoost – это реализация градиентного бустинга, которая объединяет несколько слабых моделей в сильную модель линейным образом. Что касается скорости работы, обучение модели XGBoost занимает мало времени.

В [40] исследуется применение алгоритма XGBoost в качестве модели для прогнозирования объема продаж в розничной торговле. В том числе были проанализировано влияние на точность модели других признаков, таких как погода и температура.

1.2.3 Использование методов машинного обучения для прогнозирования заболеваемости Covid-19

Применение методов машинного обучения для изучения вируса COVID-19 является необходимой мерой, которую можно использовать для сдерживания дальнейшего распространения этого заболевания. Обычные методы, используемые для прогнозирования развития COVID-19, работают медленно и дорого, а данных мало.

В статье [30] рассматриваются несколько алгоритмов машинного обучения для прогнозирования количества пациентов, заразившихся коронавирусной инфекцией. В работе применялись следующие методы: бэггинг, бустинг, метод опорных векторов, дерево решений, наивный байесовский метод, k-ближайший соседей, случайный лес и полиномиальная логистическая регрессия. Также в работе анализируется влияние фильтрации шума на производительность алгоритмов и производится сравнение работы моделей машинного обучения на зашумленных и отфильтрованных данных. В результате шумоподавления модели машинного обучения дали высокие результаты для прогнозирования случаев заболевания COVID-19 в Южной Корее. В отдельных случаях

после выполнения операций фильтрации шума методы машинного обучения достигли точности от 98 до 100%. Результаты показывают, что фильтрация шума из набора данных может повысить точность алгоритмов прогнозирования случаев COVID-19.

В статье [20] авторы рассматривают несколько подходов для прогнозирования подтвержденных случаев заболевания вирусом COVID-19, смертельных исходов, а также количества выздоровевших людей. Для поставленной задачи использовались следующие алгоритмы машинного и глубокого обучения: случайный лес, дерево решений, k-ближайших соседей, регрессия Лассо, линейная регрессия, байесовская регрессия, XGBoost, модель Хольта-Винтерса, Facebook Prophet, LSTM и т.д. Алгоритм случайного леса, модель Facebook Prophet и нейронная сеть LSTM показали наилучшие результаты с высокой точностью прогноза.

1.3. Вероятностное прогнозирование временных рядов

В настоящее время работу исследователей по прогнозированию можно разделить на две категории: точечное прогнозирование и вероятностное прогнозирование. Модели точечного прогнозирования в основном включают статистические модели и модели на основе машинного обучения.

Поскольку точечный прогноз может дать только детерминированный результат, он содержит ограниченную информацию о случайном и флуктуирующем процессе. Поэтому исследователи стали обращать внимание на вероятностное прогнозирование. При определенном уровне достоверности вероятностное прогнозирование может создавать интервалы прогнозирования, чтобы предоставить более неопределенные данные.

Автор статьи [37] использовал гауссовский процесс в качестве непараметрической модели для получения вероятностных результатов. В гауссовском процессе подразумевается предположение о нормальном распределении, но это предположение не обязательно применимо ко всем сценариям

прогнозирования вероятности. Кроме того, некоторые ученые предлагают методы, основанные на квантильной регрессии. В работе [38] использовалось экстремальное машинное обучение и квантильная регрессия, чтобы создать модель прогнозирования фотоэлектрических интервалов для измерения неопределенности и изменчивости фотоэлектрической мощности. Однако прогнозирование вероятности основано на результатах каждого квантиля, и модель необходимо обучать на каждом квантиле. Поэтому сложность модели очень высока при низкой скорости работы.

В статье [22] предлагается вероятностная модель прогнозирования солнечной радиации на основе алгоритма XGBoost. Поскольку XGBoost получается путем минимизации остатков последовательных итераций нескольких деревьев, при прогнозировании солнечной радиации в определенное время в будущем эти деревья могут итеративно генерировать несколько прогнозируемых значений освещенности. Метод оценки плотности ядра (KDE) применяется для преобразования приведенных выше результатов прогнозирования в интервалы вероятностного прогнозирования при различных уровнях достоверности. Экспериментальные результаты показывают, что этот метод по сравнению с другими эталонными алгоритмами имеет более высокую точность и дает самый узкий интервал прогнозирования с учетом уровня достоверности. Эксперимент также показывает, что метод, предложенный в этой статье, требует меньше времени на обучение и простой настройки параметров, что очень удобно для применения в производстве.

На текущий момент исследований на тему вероятностного прогнозирования развития пандемии Covid-19 с использованием алгоритмов искусственного интеллекта не найдено.

1.4. Постановка цели и задачи

Целью данной работы является вероятностное прогнозирование развития пандемии Covid-19 на основе исторических данных по заболеваемости и статистики по вакцинации.

Основные задачи, которые необходимо решить для достижения данной цели:

- поиск подходящих датасетов;
- прогнозирование количества людей, зараженных Covid-19, на основе исторических данных по заболеваемости;
- исследование влияния вакцинации на заболеваемость;
- прогнозирование количества людей, зараженных Covid-19, в зависимости от количества вакцинированных людей с использованием различных алгоритмов машинного обучения;
- вероятностное прогнозирование количества людей, зараженных Covid-19, в зависимости от количества вакцинированных людей с использованием различных алгоритмов машинного обучения.
- сравнение моделей;

ГЛАВА 2. МАТЕМАТИЧЕСКОЕ ОПИСАНИЕ МОДЕЛЕЙ

Временной ряд – это ряд наблюдений, перечисленных в порядке времени. Точки данных во временном ряду обычно записываются через постоянные последовательные интервалы времени.

Анализ временных рядов – это процесс извлечения значимой нетривиальной информации и закономерностей из временных рядов.

Прогнозирование временных рядов – это процесс предсказания будущих значений временного ряда на основе прошлых наблюдений и других входных данных.

Пример временного ряда с еженедельным суммарным количеством заболеваний вирусом Covid-19 в штате Аляска за период с начала 2020-го года до марта 2023 года показан на рис. 2.1.

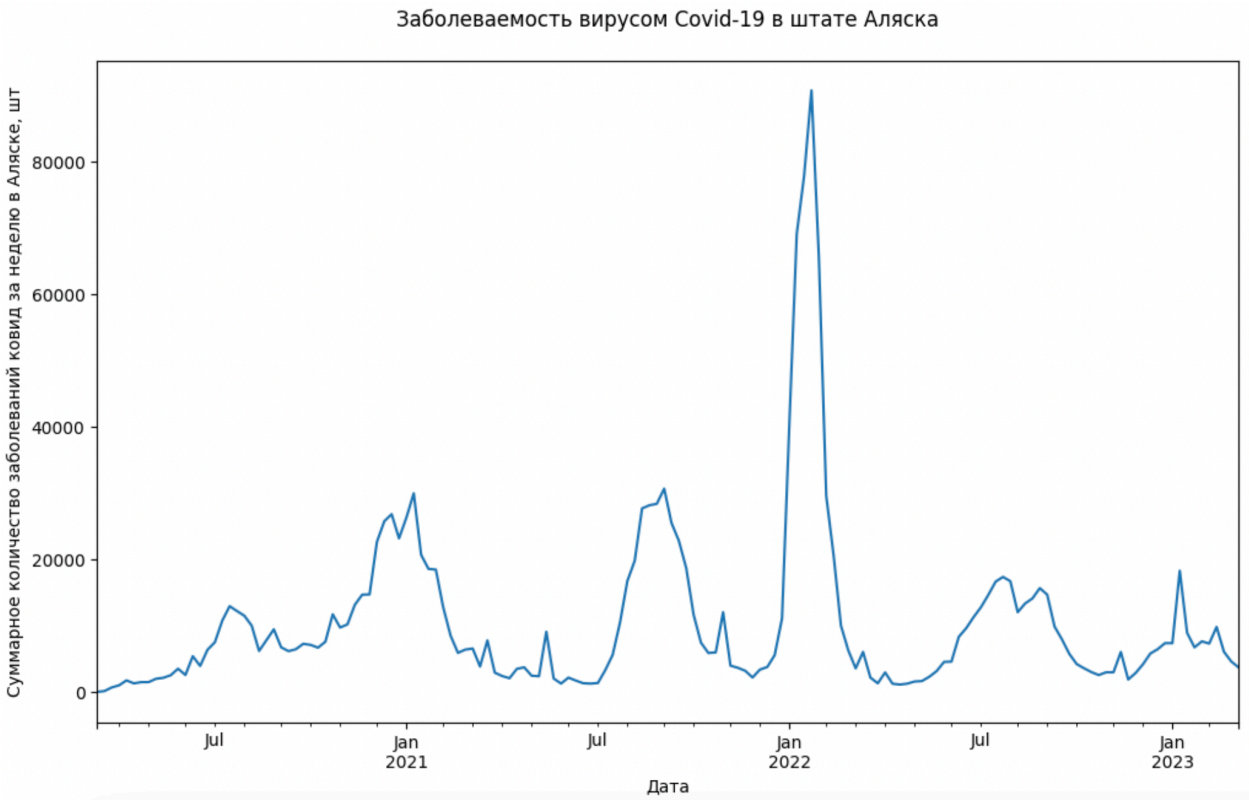

Рис. 2.1. Еженедельное суммарное количество заболеваний вирусом Covid-19 в штате Аляска

Цель прогнозирования временных рядов состоит в том, чтобы использовать историческую информацию о конкретном значении целевой переменной и делать прогнозы значений той же целевой переменной в будущем.

В целом, между прогнозированием временных рядов и другими прогностическими моделями машинного обучения с учителем есть два важных отличия.

Во-первых, время является важным предиктором во многих сферах. При предсказании временных рядов речь идет о прогнозировании конкретной переменной, учитывая, что известно, как эта переменная менялась с течением времени в прошлом. В других прогностических моделях (например, задачи регрессии), временная составляющая данных игнорируется или недоступна. Такие данные известны как данные поперечного сечения (cross-sectional data) или поперечный разрез исследуемой совокупности. Они собираются путем наблюдения за многими объектами в один период времени, как показано на рис 2.2.

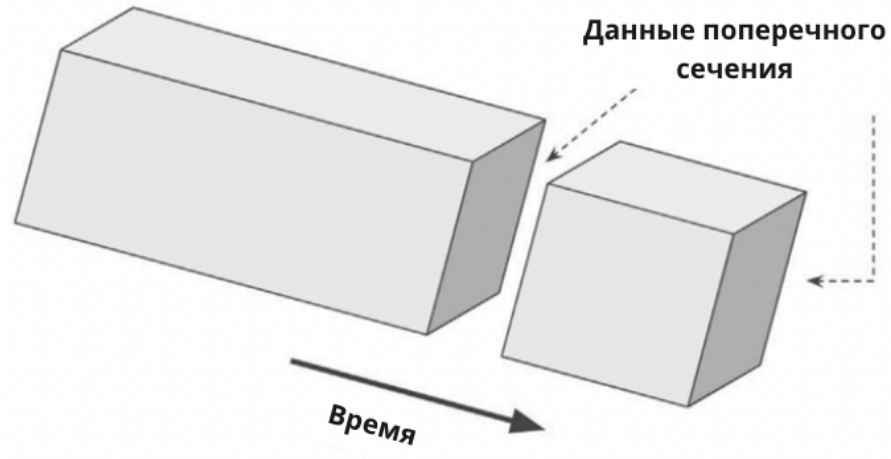

Рис. 2.2. Данные поперечного сечения в подмножестве данных временного ряда

Во-вторых, можно не использовать или даже не иметь данных для других переменных, которые потенциально могут повлиять на целевую переменную. Таким образом, независимые переменные или переменные-предикторы не являются строго необходимыми для прогнозирования одномерных

временных рядов, но их рекомендуется использовать для многомерных временных рядов.

2.1. Описание статистических моделей прогнозирования временных рядов

2.1.1. ARIMA

Модели ARIMA – это класс моделей, способных прогнозировать как стационарные, так и нестационарные временные ряды и давать точные прогнозы на основе описания исторических данных. Поскольку данная модель не предполагает какой-либо конкретной закономерности в исторических данных временного ряда, который должен быть спрогнозирован, эта модель сильно отличается от других моделей, используемых для прогнозирования.

Модель ARIMA характеризуется 3 переменными [31] и обычно обозначается как ARIMA(p,q,d):

- p – порядок авторегрессии (AR), который позволяет добавить предыдущие значения временного ряда;
- d – порядок интегрирования (I), который указывает количество производных, необходимых для формирования стационарного временного ряда;
- q – порядок скользящего среднего (MA), который позволяет установить погрешность модели как линейную комбинацию наблюдавшихся ранее значений ошибок.

Вместе эти три параметра учитывают сезонность, тенденцию и шум в наборах данных.

Модель регрессии, которая использует свои задержки в качестве предикторов, называется «авторегрессивной» [6]. В случае моделей авторегрессии выходными данными являются будущие точки данных, которые могут быть выражены как линейная комбинация прошлых p точек данных, где p – окно

задержки. Авторегрессионная модель с p задержками или модель AR(p) выражается следующим уравнением:

$$y_t = \alpha + \beta_1 y_{t-1} + \beta_2 y_{t-2} \ldots + \beta_p y_{t-p} + \varepsilon_1, \quad (2.1)$$

где y_{t-1} – 1 запаздывание (lag) временного ряда;

β_1 – коэффициент 1 задержки (lag), который оценивает модель;

α – срок перехвата, также оцененный моделью.

ε_1 – шум.

В модели AR(p) запаздывающий ряд – это новый предиктор, используемый для подбора зависимой переменной, которая по-прежнему является исходным значением ряда y_t.

Нестационарный временной ряд можно преобразовать в стационарный временной ряд с помощью метода, называемого дифференцированием. Дифференцированный ряд – это изменение между последовательными точками данных в ряду. Дифференцирование первого порядка описывается выражением:

$$y'_t = y_t - t_{t-1}$$

В некоторых случаях однократное дифференцирование все равно дает нестационарный временной ряд. В этом случае требуется дифференцирование второго порядка. Дифференцирование второго порядка – это изменение между двумя последовательными точками данных во дифференцированном временном ряду первого порядка.

Подводя итог, дифференцирование порядка d используется для преобразования нестационарных временных рядов в стационарный временной ряд.

$$y'_t = y_t - t_{t-d}$$

В дополнение к созданию регрессии фактических прошлых значений p, как показано в уравнении (2.1), можно также создать уравнение регрессии, включающее ошибки прогноза прошлых данных, и использовать его в качестве предиктора. В данном случае y_t зависит только от запаздывающих ошибок прогноза. Это имеет смысл для прошлых точек данных, но не для точки

данных t, потому что она все еще прогнозируется. Следовательно, ε_t считается белым шумом. Уравнение регрессии для y_t можно понимать как взвешенное (с весом φ) скользящее среднее прошлых q ошибок прогноза. Уравнение для модели скользящей средней с q задержками или MA(q) выражается следующим образом:

$$y_t = I + \varepsilon_t + \varphi_1 \varepsilon_{t-1} + \varphi_2 \varepsilon_{t-2} + .. + \varphi_q \varepsilon_{t-q},$$

где ε_i– это ошибки прогноза моделей соответствующих задержек i.

Ошибки ε_t и ε_{t-1} вычисляются на основе следующих выражений:

$$y_t = \beta_1 y_{t-1} + \beta_2 y_{t-2} \ldots + \beta_0 y_0 + \varepsilon_t$$

$$y_{t-1} = \beta_1 y_{t-2} + \beta_2 y_{t-3} \ldots + \beta_0 y_0 + \varepsilon_{t-1}$$

Модель авторегрессионного интегрированного скользящего среднего (ARIMA) представляет собой комбинацию дифференциальной авторегрессионной модели с моделью скользящего среднего. Это выражается как:

$$y'_t = I + \beta_1 y'_{t-1} + \beta_2 y'_{t-2} + .. + \beta_n y'_{t-n} + \varepsilon_t + \varphi_1 \varepsilon_{t-1} + \varphi_2 \varepsilon_{t-2} + .. + \varphi_n \varepsilon_{t-n} \quad (2.2)$$

Часть AR показывает, что временной ряд регрессирует на свои собственные прошлые данные. Часть MA указывает, что ошибка прогноза представляет собой линейную комбинацию прошлых соответствующих ошибок. Часть I показывает, что значения данных были заменены дифференциальными значениями порядка d для получения стационарных данных, что является требованием подхода модели ARIMA.

Оценка коэффициентов β и φ для заданных p, d, q — это то, что делает ARIMA, когда он учится на данных обучения во временном ряду [3]. Указание p, d, q может быть сложным ограничением, но можно попробовать различные комбинации и оценить производительность модели. Как только модель ARIMA указана со значением p, d, q, коэффициенты уравнения (2.2) необходимо оценить. Наиболее распространенный способ оценки – оценка максимального правдоподобия. Это похоже на оценку наименьших квадратов для уравнения регрессии, за исключением того, что MLE находит коэффициенты

модели таким образом, чтобы максимизировать шансы найти фактические данные.

Модель ARIMA можно дополнительно улучшить, чтобы учесть сезонность во временном ряду. Сезонная модель ARIMA выражается понятием ARIMA(p,d,q)(P,D,Q)m, где:

- p – порядок несезонной авторегрессии;
- d – порядок интегрирования;
- q – порядок несезонной скользящей средней ошибки;
- P – порядок сезонной авторегрессии;
- D – порядок сезонного интегрирования;
- Q – порядок сезонной скользящей средней ошибки.
- m – количество наблюдений в году (для годовой сезонности).

Сезонная часть ARIMA аналогична терминам, используемым в несезонной части, за исключением того, что она сдвигается назад m раз, где m – период сезонности.

2.1.2. Экспоненциальное сглаживание

Экспоненциальное сглаживание – это средневзвешенное значение прошлых данных, при этом последним точкам данных придается больший вес, чем более ранним точкам данных. Веса экспоненциально затухают по направлению к более ранним точкам данных, отсюда и название.

Взвешенное среднее описывается уравнением:

$$F_{n+1} = \sum_{n=1}^{k} \omega_n y_t + 1 - n$$

Если вместо взвешивания последних n значений ряда взвешивать все доступные наблюдения, при этом экспоненциально уменьшая веса по мере углубления в исторические данные, то получаем экспоненциальное сглаживание:

$$F_{n+1} = \alpha\, y_n + \alpha\,(1-\alpha)\, y_{n-1} + \alpha\,(1-\alpha)^2 y_{n-2} + \ldots, \qquad (2.3)$$

где α – вес, который обычно находится в диапазоне от 0 до 1.

Заметим, что $\alpha = 1$ возвращает наивный прогноз уравнения. Использование более высокого значения α приводит к тому, что недавним значениям придается больший вес, и результирующая кривая становится ближе к фактической кривой, но использование более низкого значения α приводит к тому, что больше внимания уделяется ранее прогнозируемым значениям, что приводит к более гладкой, но менее точным прогнозируемым значениям.

Чтобы прогнозировать будущие значения с помощью экспоненциального сглаживания, уравнение (2.3) можно переписать как:

$$F_{n+1} = \alpha\, y_n + (1 - \alpha) F_n \tag{2.4}$$

$$F_n = \alpha\,(\, y_{n-1} + \alpha\,(1 - \alpha)\, y_{n-2} + ..)$$

Уравнение (2.4) более полезно, поскольку оно включает в себя как фактическое значение y_n, так и прогнозируемое значение F_n. Более высокое значение α дает точную подгонку, а более низкое значение дает более плавную подгонку. Это концепция простого экспоненциального сглаживания.

Простое экспоненциальное сглаживание является основой для нескольких распространенных методов прогнозирования на основе сглаживания. Модель простого экспоненциального сглаживания подходит только для временных рядов без четкого тренда или сезонности. Такая модель имеет только один параметр, α, и может помочь сгладить данные во временном ряду, чтобы их можно было легко экстраполировать и делать прогнозы.

Кроме того, из уравнения (2.3), можно увидеть, что прогнозы нельзя делать более чем на один шаг вперед, потому что для прогноза для шага $(n + 1)$ нужны данные для предыдущего шага n. Для составления долгосрочных прогнозов, т.е. где горизонт прогнозирования $h \gg 1$, необходимо также учитывать информацию о тренде и сезонности. Как только тренд и сезонность зафиксированы, можно прогнозировать значение на любое время в будущем, а не только значения на один шаг вперед.

Упрощенная модель экспоненциального сглаживания, описанная ранее, не особенно эффективна для выявления трендов. Для этого необходимо расширение этой техники, называемое двойным экспоненциальным сглаживанием Хольта.

Экспоненциальное сглаживание (2.3) просто вычисляет среднее значение временного ряда при $n + 1$. Если ряд также имеет тренд, то необходимо также оценить средний наклон ряда. Это то, что делает двойное сглаживание Хольта с помощью другого параметра, β. Уравнение сглаживания, подобное уравнению (2.3) построено для среднего тренда при $n + 1$. С двумя параметрами, α и β, любой временной ряд с трендом может быть смоделирован и, следовательно, спрогнозирован. Прогноз может быть выражен как сумма этих двух компонентов, среднего значения или «уровня» ряда (или ожидаемое значение ряда), Ln, и тренда, Tn, рекурсивно как:

$$F_{n+1} = L_n + T_n \,, \tag{2.4}$$

где:

$$L_n = \alpha\, y_n + (1 - \alpha)(L_{n-1} + T_{n-1})$$
$$T_n = \beta\,(L_n + L_{n-1}) + (1 - \beta)\, T_{n-1}$$

В результате получаем набор функций. Первая описывает уровень, который зависит от текущего значения ряда, а второе слагаемое теперь разбивается на предыдущее значение уровня и тренда. Вторая отвечает за тренд, который зависит от изменения уровня на текущем шаге, и от предыдущего значения тренда. Здесь в роли веса в экспоненциальном сглаживании выступает коэффициент. Наконец, итоговое предсказание представляет собой сумму модельных значений уровня и тренда.

Чтобы сделать прогнозом на некоторый горизонт прогнозирования, можно изменить уравнение (2.4) следующим образом:

$$F_{n+1} = L_n + h\, T_n$$

Значения параметра можно оценить на основе наилучшего соответствия обучающим (прошлым) данным.

Когда временной ряд содержит сезонность в дополнение к тренду, для оценки сезонного компонента временного ряда потребуется еще один параметр, γ. Оценки уровня теперь корректируются с помощью сезонного индекса, который рассчитывается с помощью третьего уравнения, включающего γ. Это называется тройным экспоненциальным сглаживанием Хольта-Винтерса и выражается следующим образом:

$$F_{t+h} = (L_t + h\,T_n)S_{t+h-p}$$

$$L_t = \frac{\alpha\, y_t}{S_{t-p}} + (1-\alpha)(L_{t-1} + T_{t-1})$$

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

$$S_t = \gamma\left(\frac{y_t}{L_t}\right) + (1-\gamma)\,S_{t-p}$$

где p – период сезонности.

Можно оценить значение параметров α, β, γ из подгонки уравнения сглаживания к обучающим данным.

2.2. Описание моделей машинного обучения для прогнозирования временных рядов

Контролируемое машинное обучение (обучение с учителем) – это метод машинного обучения, основанный на использовании наборов размеченных данных. Чтобы применить методы контролируемого обучения к данным временных рядов, используется метод оконного преобразования, который преобразует временные ряды в данные поперечного сечения [3]. Для этого набор последовательных моментов времени определяется как «окна», где самое последнее наблюдение служит целевой переменной, а предыдущие наблюдения служат входными переменными, как показано на рис. 2.3. Этот процесс с последовательными окнами повторяется, и каждая точка ряда может выступать как входной, так и выходной переменной в разное время. Как только будет создано достаточное количество окон, модель может быть обучена на основе предполагаемых взаимосвязей между запаздывающими входными данными и

целевыми выходными данными аналогично авторегрессионным моделям, которые используют прошлые p-наблюдений для прогнозирования будущих значений.

Контролируемые ученики (supervised learners) могут использоваться для изучения и прогнозирования целевой переменной, то есть следующего временного шага во временном ряду. Выведенная модель прогнозирует будущую точку данных на основе последнего окна временного ряда, обеспечивая представление одной будущей точки данных. Новая прогнозируемая точка данных может использоваться для определения нового окна и прогнозирования еще одной точки данных в будущем. Этот процесс повторяется до тех пор, пока не будут сделаны все будущие прогнозы.

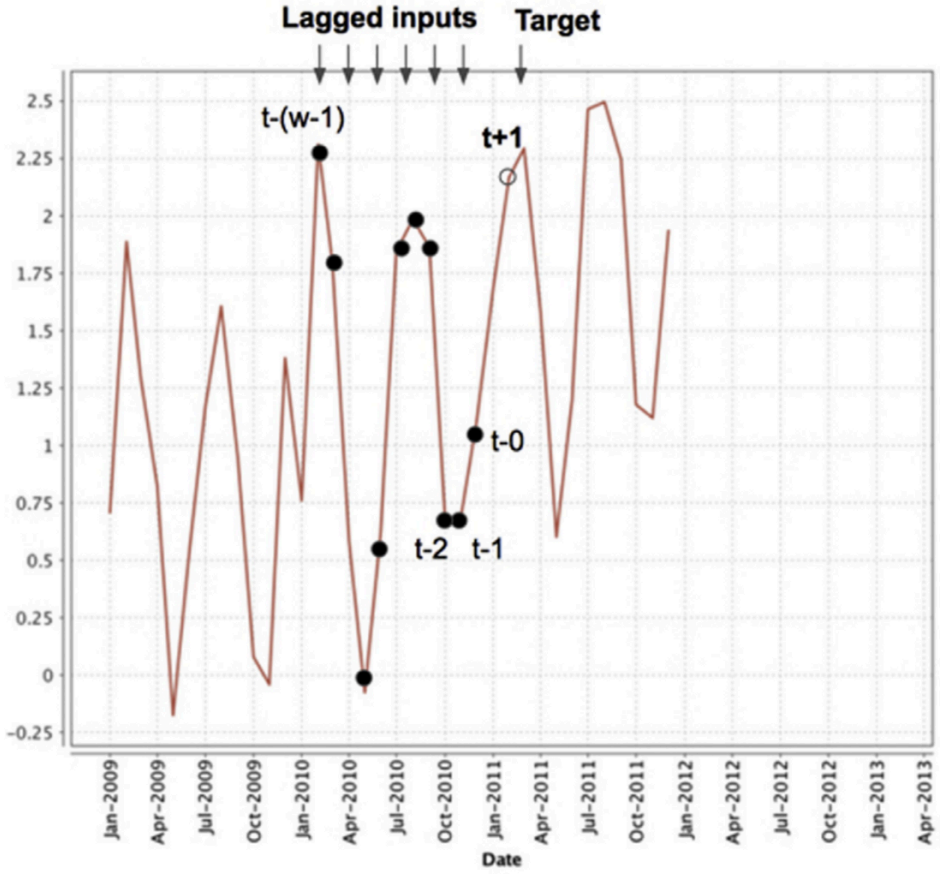

Рис. 2.3. Входные данные с задержкой и целевая переменная.

Задача регрессии является одной из задач, решаемых с помощью контролируемого машинного обучения**.** Основная идея заключается в том, чтобы предсказать значение зависимого атрибута Y путем объединения атрибутов-предикторов X в функцию: $Y = f(x)$. Подгонка функций включает в себя множество различных методов. Некоторые из них будут рассмотрены в данном разделе.

2.2.1. Алгоритм k-ближайших соседей

Метод k-ближайших соседей (k Nearest Neighbors, или kNN) – популярный алгоритм для решения задач классификации и регрессии. На интуитивном уровне суть метода проста: посмотри на соседей вокруг, какие из них преобладают, таковым ты и являешься

В регрессии каждый пример состоит из вектора функций, описывающих пример, и связанного с ним числового целевого значения. Учитывая новый пример, KNN находит его k наиболее похожих примеров, называемых ближайшими соседями, в соответствии с заданной метрикой расстояния, например, евклидово расстояние, и прогнозирует его значение как совокупность целевых значений, связанных с его ближайшими соседями. В случае использования метода для регрессии, объекту присваивается среднее значение по k ближайшим к нему объектам, значения которых уже известны.

В качестве метрики расстояния может использоваться Евклидово расстояние, расстояние Минковского, расстояние Чебышева и т.д. Евклидово расстояние вычисляется по формуле:

$$\sqrt{\sum_{x=1}^{n}(f_x^i - q_x)^2}$$

Точки данных, находящиеся ближе друг к другу, похожи, а более дальние соседи должны иметь меньшее влияние на определение окончательного результата. Чтобы учесть влияние точек данных всем соседям назначаются

веса ω_i, причем веса увеличиваются по мере приближения соседей к контрольной точке данных [3]. Веса включаются на заключительном этапе прогнозирования. Веса ω_i должны удовлетворять двум условиям: они должны быть пропорциональны расстоянию тестовой точки данных от соседа, а сумма всех весов должна быть равна единице. Расчет весов выражается экспоненциальным затухание в зависимости от расстояния и вычисляется с помощью уравнения:

$$\omega_i = \frac{e^{-d(x,n_i)}}{\sum_{i=1}^{k} e^{-d(x,n_i)}},$$

где ω_i – вес i -го соседа n_i,

k – итоговое количество соседей,

x – значение из тестового набора данных.

Вес используется для предсказания целевой переменной y':

$$\hat{y} = mean\ (w_1 * y_1, w_2 * y_2, \dots w_k * y_k),$$

где y_i – класс, предсказанный соседом n_i.

2.2.2. Ансамблевые модели

Методы ансамбля оптимизируют проблему поиска наилучшей модели для описания взаимосвязи в конкретной обучающей выборки данных с наименьшей ошибкой. Для этого используется набор отдельных моделей прогнозирования, а затем эти модели комбинируются для формирования совокупной модели. Эти методы обеспечивают технику для создания лучшей модели путем объединения нескольких моделей в одну, что значительно повышает точность.

Модели ансамбля имеют набор базовых моделей, так называемых «слабых учеников», которые принимают одни и те же входные данные и независимо предсказывают результат. Затем выходные данные всех этих базовых моделей объединяются путем усреднения для формирования ансамблевого вывода и создают одного «сильного ученика».

Выделяют 3 основных ансамблевыми метода: стекинг, бэггинг, бустинг.

Бэггинг – это метод, при котором базовые модели разрабатываются путем изменения обучающего набора для каждой базовой модели. В заданном обучающем наборе T из n записей создаются m обучающих наборов, каждый из которых содержит n записей, путем выборки/сэмплирования с заменой. Каждый обучающий набор $T_1, T_2, T_3, .. T_m$ будет иметь такое же количество записей n, что и исходный обучающий набор T. Поскольку они отбираются с замещением, они могут содержать повторяющиеся записи. Это называется бутстрэппинг (bootstrapping). Каждая выборка из обучающего набора затем используется для подготовки базовой модели. Благодаря бутстрэппингу уже имеется набор из m базовых моделей. Далее прогноз каждой модели агрегируется для модели ансамбля. Эта комбинация бутстрэппинг и агрегирования называется бэггингом (рис 2.4).

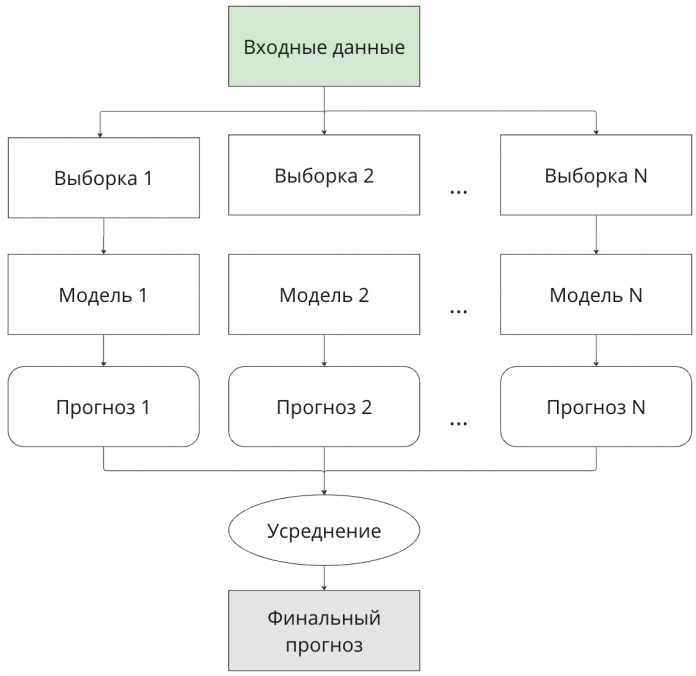

Рис 2.4. Концепция бэггинга

Бустинг является еще один подходом к построению модели ансамбля. В отличие от бэггинга, бустинг обучает базовые модели итеративно и последовательно одну за другой и присваивает веса всем обучающим записям.

Процесс бустинга концентрируется на обучающих данных, которые трудно предсказывать, и представляет их в обучающем наборе для следующей итерации. Обучающие выборки выбирается на основе веса и затем используется для построения модели. Неправильно предсказанным данным присваивается более высокий вес, поэтому записи, которые трудно классифицировать, имеют более высокую вероятность выбора для следующей итерации. В результате формируется ансамбль базовых учеников, специализирующихся на предсказании как легко и трудно прогнозируемых записей. Далее все базовые ученики объединяются. Концепция бустинга показана на рис. 2.5.

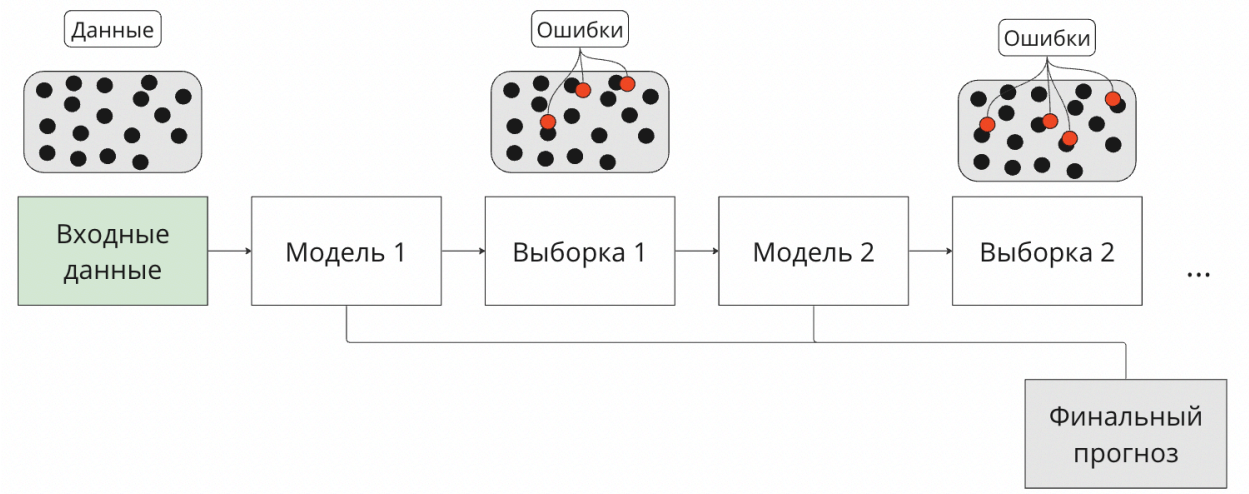

Рис 2.5. Концепция бустинга

2.2.2.1 Случайный лес

Алгоритм случайного леса (Random Forest) – алгоритм машинного обучения, который строится на основе ансамбля решающих деревьев.

Дерево решений – это алгоритм машинного обучения, который стоится на основе иерархического объединения логических правил вида «если .., то ..».

Метод случайного леса строится на основе концепции, используемой в бэггинге. При принятии решения о разделении каждого узла в дереве решений случайный лес рассматривает только случайное подмножество атрибутов в

обучающем наборе. Алгоритм является случайным на двух уровнях: выбор обучающей выборки и выбор атрибута во внутренней работе каждой базовой модели [3]. Концепция случайных лесов впервые была предложена Лео Брейманом и Адель Катлер. В целом модель работает, используя следующие шаги [7]. Если имеется n обучающих записей с m атрибутами и k количеством деревьев в лесу; для каждого дерева:

1. Выбирается случайная выборка размера n с замещением.
2. Выбирается количество атрибутов, которые следует учитывать при разделении узла D, где $D \ll m$.
3. Запускается алгоритма дерева решений. При этом для каждого узла вместо рассмотрения всех m атрибутов рассматривается D атрибутов. Этот шаг повторяется для каждого узла.
4. Как только все деревья в лесу построены и для каждой новой записи все деревья прогнозируют результат. Усредненный итоговый результат по всем базовым деревьям – это предсказание леса.

2.2.2.2 Градиентный бустинг

Градиентный бустинг – это метод машинного обучения, представляющий собой линейную аддитивную модель, состоящую из ансамбля слабых моделей прогнозирования [40].

Алгоритм экстремального повышения градиента или XGBoost – это эффективный и быстрый алгоритм дерева решений. XGBoost представляет собой реализацию градиентного бустинга, которая объединяет несколько слабых учеников в одного сильного ученика линейным образом. Что касается скорости работы, XGBoost поддерживает параллельный выбор точек разделения, а обучение модели требует гораздо меньше времени.

Основная идея XGBoost заключается в постоянном добавлении слабых деревьев с разным весом в набор. Деревья в наборе должны максимально

приближаться к остаткам предыдущего прогноза, что выражается следующим образом:

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), \quad f_k \in F \ ,$$

где $\hat{y}_i$ – предсказанное значение;

F – множество, включающее все деревья регрессии;

f_k – одно из деревьев регрессии;

K – количество деревьев регрессии.

Ожидается, что прогнозируемое значение $\hat{y}_i$ будет как можно ближе к истинному значению y_k и при этом не потеряет своей способности к обобщению. Формула для вычисления целевой функции Obj приведена ниже.

$$Obj^{(t)} = \sum_{i=1}^{n} L\left(y_i, \hat{y}_i^{(t)}\right) + \sum_{i=1}^{t} \Omega(f_t) + constant \ ,$$

где $L\left(y_i, \hat{y}_i^{(t)}\right)$ – функция потерь, которая представляет собой разницу между прогнозируемым значением и истинным значением. Это может быть любая форма функции потерь, которая является производной второго порядка.

$\Omega(f_t)$ – функция регуляризации, определяющая сложность модели, которая описывается выражением:

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda ||\omega||^2,$$

где T – число листовых узлов,

ω – оценка, представленная листовыми узлами.

Чем меньше значение $\Omega(f_t)$, тем ниже сложность и выше способность к обобщению.

XGBoost использует разложение Тейлора второго порядка для расширения функции потерь в процессе повышения градиента. Окончательная целевая функция имеет следующий вид

$$Obj^{(t)} =\cong \sum_{i=1}^{n}\left[g_i\, f_t(x_i) + \frac{1}{2} h_i\, f_t^2(x_i)\right] + \Omega(f_t)$$

$$= \sum_{i=1}^{n}\left[g_i\, \omega_{q(x_i)} + \frac{1}{2} h_i\, \omega_{q(x_i)}^2\right] + \gamma\, T + \frac{1}{2}\,\lambda \sum_{j=1}^{T} \omega_j^2$$

$$= \sum_{j=1}^{T}\left[\left(\sum\nolimits_{i\in I_j} g_i\right)\omega_j + \frac{1}{2}\left(\sum\nolimits_{i\in I_j} h_i + \lambda\right)\omega_j^2\right] + \gamma\, T$$

где $g_i = \partial_{\hat{y}_i{}^{(t-1)}} L\left(y_i, \hat{y}_i{}^{(t-1)}\right)$ – производная первого порядка каждой точки данных в функции ошибок;

$h_i = \partial^2_{\hat{y}_i{}^{(t-1)}} L\left(y_i, \hat{y}_i{}^{(t-1)}\right)$ – производная второго порядка каждой точки данных в функции ошибок;

I_j – набор индексов выборок на каждом листовом узле j:

$$I_j = \{i | q(x_i) = j\}$$

Для данного $q(x_i)$, взяв производную от w_j равной 0, можно получить наилучший вес w_j^* листового узла j.

$$w_j^* = -\frac{\sum_{i\in I_j} g_i}{\sum_{i\in I_j} h_i + \lambda}$$

Таким образом, XGBoost использует методологию оптимизации градиентного спуска и произвольных дифференцируемых функций потерь для минимизации функции потерь путем добавления слабых учеников, т. е. для определения и оптимизации целевой функции. Чтобы минимизировать целевую функцию, XGBoost использует жадный алгоритм для поиска оптимальной древовидной структуры [24].

2.3 Вероятностное прогнозирование

Условные модели описывают условное распределение $F_{Y|x}(y|x)$ исхода Y, зависящего от наблюдаемых признаков x. Вместо моделирования полного распределения $Y|x$ большинство моделей прогнозирования фокусируются на

предсказании наиболее вероятного значения, например, среднего значения результата [32]. Это называется точечным прогнозированием.

Вероятностное прогнозирование, в отличие от точечного прогнозирования, представляет собой семейство методов, которые позволяют прогнозировать ожидаемое распределение результата, а не одно будущее значение. Этот тип прогнозирования предоставляет гораздо более богатую информацию, поскольку он сообщает диапазон вероятных значений, в который может попасть истинное значение, что позволяет оценить интервалы прогнозирования.

Интервал прогнозирования – это интервал, в пределах которого ожидается, что истинное значение переменной отклика будет найдено с заданной вероятностью. В общем случае интервал прогнозирования можно записать как

$$\hat{y}_{t+h|T} + c\widehat{\sigma}_h,$$

где $\widehat{\sigma}_h$ - оценка стандартного отклонения шага h распределения прогноза, а множитель c зависит от вероятности покрытия

В случае нормального распределения для 95-процентный интервал прогнозирования коэффициент $c = 1.96$.

Существует несколько способов оценки интервалов прогнозирования, большинство из которых требуют, чтобы остатки (ошибки) модели подчинялись нормальному распределению. Однако, когда это предположение невозможно сделать, обычно используются две альтернативы: бутстрэппинг и квантильная регрессия.

Общей чертой интервалов прогнозирования является то, что они обычно увеличиваются по мере увеличения горизонта прогнозирования. Чем дальше мы прогнозируем, тем больше неопределенности связано с прогнозом и, следовательно, тем шире интервалы прогнозирования. То есть, σ_h обычно увеличивается с h. Но есть некоторые нелинейные методы прогнозирования, не обладающие этим свойством.

Для получения интервала прогнозирования необходимо иметь оценку σ_h. Для одношаговых прогнозов ($h = 1$) оценка стандартного отклонения прогноза σ_1 вычисляется с помощью уравнения:

$$\widehat{\sigma} = \sqrt{\frac{1}{T-K-M}\sum_{t=1}^{T} e_t^2},$$

где K – количество параметров, оцененных в методе прогнозирования,

M – количество пропущенных значений в остатках.

Для многошаговых прогнозов требуется более сложный метод расчета. Эти расчеты предполагают, что остатки не коррелированы.

Также стоит отметить разницу интервала прогнозирования и доверительного интервала. Интервал прогнозирования предсказывает, в какой диапазон попадет будущее отдельное наблюдение, а доверительный интервал показывает вероятный диапазон значений, связанных с некоторым статистическим параметром данных, таким как среднее значение генеральной совокупности.

ГЛАВА 3. РЕАЛИЗАЦИЯ МОДЕЛЕЙ ПРОГНОЗИРОВАНИЯ

3.1. Описание данных и анализ данных

Данные статистики по заболеваемости и статистики по вакцинации для исследования взяты с сайта Университета Джонса Хопкинса [43].

Статистика по заболеваемости включает в себя данные случаев заболевания вирусом Covid-19 в Соединенных Штатах Америки по каждому штату с января 2020 года. Данные по заболеваемости были предварительно преобразованы и очищены от лишних атрибутов. Датасет представлен на рис 3.1 и состоит из трех столбцов: даты, штата и количество случаев заболевания вирусом в каждом штате ежедневно. Количество случаев заболевания представлено в двух вариантах: оригинальное количество ежедневно и кумулятивная сумма.

	state_abbr	covid_cases	orig_covid_cases
date			
2020-01-22	AK	0	0.0
2020-01-22	AL	0	0.0
2020-01-22	AR	0	0.0
2020-01-22	AZ	0	0.0
2020-01-22	CA	0	0.0
...	...	...	...
2023-03-09	VT	152618	0.0
2023-03-09	WA	1928913	0.0
2023-03-09	WI	2006582	708.0
2023-03-09	WV	642760	0.0
2023-03-09	WY	185385	0.0

57150 rows × 3 columns

Рис. 3.1. Данные по заболеваемости коронавирусом в США по штатам.

Также использовались данные по вакцинации от коронавируса Covid-19, показанные на рис 3.2. Данные представляют собой ежедневную статистику вакцинации по каждому штату. Вакцинация характеризуется двумя атрибутами: количеством людей, вакцинированных хотя бы одной дозой и

количеством людей, прошедших полную вакцинацию. Оба поля представлены в двух вариантах: оригинальное количество ежедневно и кумулятивная сумма.

date	state_abbr	people_at_least_one_dose	people_fully_vaccinated	orig_people_at_least_one_dose	orig_people_fully_vaccinated
2021-02-14	AK	122518	0	122518.0	0.0
2021-02-15	AK	126444	0	3926.0	0.0
2021-02-16	AK	128412	0	1968.0	0.0
2021-02-17	AK	129734	0	1322.0	0.0
2021-02-18	AK	130830	0	1096.0	0.0
...	...	...	...	...	...
2023-03-01	WY	352618	306903	0.0	0.0
2023-03-02	WY	352696	306951	78.0	48.0
2023-03-03	WY	352696	306951	0.0	0.0
2023-03-04	WY	352696	306951	0.0	0.0
2023-03-05	WY	352696	306951	0.0	0.0

37500 rows × 5 columns

Рис. 3.2. Данные по вакцинации от коронавируса в США по штатам

Помимо этого, использовались данные по летальным исходам от коронавируса Covid-19, показанные на рис 3.3. Данные представляют собой ежедневную статистику летальных исходов по каждому штату. Смертность характеризуется одним атрибутом: количеством людей, умерших от Covid-19. Данное поле представлено в двух вариантах: оригинальное количество ежедневно и кумулятивная сумма.

date	state_abbr	deaths_cases	orig_deaths_cases
2020-01-22	AK	0	0.0
2020-01-23	AK	0	0.0
2020-01-24	AK	0	0.0
2020-01-25	AK	0	0.0
2020-01-26	AK	0	0.0
...	...	...	...
2023-03-05	WY	2002	0.0
2023-03-06	WY	2002	0.0
2023-03-07	WY	2004	2.0
2023-03-08	WY	2004	0.0
2023-03-09	WY	2004	0.0

57150 rows × 3 columns

Рис. 3.2. Данные по смертности от коронавируса в США по штатам

Для использования данных для прогнозирования ежедневные временные ряды преобразованы в еженедельные, а все атрибуты при группировке данных по неделям – просуммированы.

3.2. Анализ влияния вакцинации на заболеваемость и смертность

Взаимная корреляция (кросс-корреляция) – это способ измерения степени сходства между временным рядом и запаздывающей версией другого временного ряда.

Этот тип корреляции полезен для расчета, потому что он может сказать нам, предсказывают ли значения одного временного ряда будущие значения другого временного ряда. Другими словами, он может сказать нам, является ли один временной ряд опережающим индикатором для другого временного ряда.

Взаимная корреляция определяется выражением:

$$\rho_{xy}(l) = \frac{r_{xy}(l)}{\sqrt{r_{xx}(0)}\sqrt{r_{yy}(0)}} \quad (3.1),$$

где $\sqrt{r_{xx}(0)}$ – квадратный корень из коэффициента автокорреляции x при нулевом лаге,

$\sqrt{r_{yy}(0)}$ – квадратный корень из коэффициента автокорреляции y при нулевом лаге.

Автокорреляция – это временной ряд, взаимно коррелированный сам с собой.

Знаменатель в уравнении (3.1) всегда больше числителя. Таким образом, $\rho_{xy}(l)$ не превышает 1. Уравнении (3.1) становится отрицательным, только когда кривые временных рядов имеют обратную зависимость.

Обычно используются вариант взаимной корреляции, который выражается следующим образом

$$\rho_{xy}(l) = \frac{\sum_{i=0}^{N-1}(x_i-\bar{x})*(y_{i-l}-\bar{y})}{\sqrt{\sum_{i=0}^{N-1}(x_i-\bar{x})^2}\sqrt{\sum_{i=0}^{N-1}(y_{i-l}-\bar{y})^2}} = \frac{cov(x,y)}{\sqrt{\sigma_x^2\,\sigma_y^2}} \quad (3.2)$$

Уравнение (3.2) также называется корреляцией Пирсона.

Существует эмпирическое правило – вычислять лаги только до $\frac{N}{2}$, где N – количество наблюдений во временном ряду. Также на практике часто бывает, что интерес представляют только малое количество задержек (лагов).

3.3. Прогнозирование

3.3.1 Используемые библиотеки

Для реализации поставленной задачи будет использоваться язык программирования Python. Python – это язык программирования, который широко используется в интернет-приложениях, разработке программного обеспечения, науке о данных и машинном обучении.

Python стал одним из основных продуктов в науке о данных, так как содержит готовые библиотеки для проведения сложных статистических расчетов, создания визуализаций, построения алгоритмов машинного обучения, обработки и анализа данных, а также выполнения других задач, связанных с данными. Одними из таких библиотек, используемых для анализа и прогнозирования данных временных рядов, являются sktime, sklearn, skforecast и statsmodels.

Библиотека sktime предназначена для прогнозирования временных рядов и включает в себя различные алгоритмы прогнозирования, в том числе статистические методы и модели машинного обучения. Например, одним из подходов к прогнозированию может быть использование регрессионной модели, в которой явно учитывается временное измерение данных. Другой подход заключается в том, чтобы свести проблему прогнозирования к задаче регрессии данных поперечного сечения, где входные данные представляются в виде таблиц, а задержки данных временного ряда обрабатываются как независимые функции в алгоритмах регрессии в стиле библиотеки scikit-learn. Оба этих подхода рассматривались в предыдущей главе.

Данные временного ряда относятся к данным, в которых переменные упорядочены во времени, или к индексу, указывающему положение наблюдения в последовательности значений. В sktime данные временных рядов могут относиться к данным, которые являются одномерными, многомерными или панельными.

Scikit-learn – это библиотека Python, которая предоставляет множество алгоритмов машинного обучения с учителем и без. Функциональность, предоставляемая scikit-learn, включает в себя: регрессию, классификацию, кластеризацию, выбор подходящей модели и предварительная обработку данных.

Библиотека skforecast предназначена для использования регрессоров scikit-learn в качестве прогнозистов для временных рядов. Помимо этого, skforecast также есть возможность работы с любым регрессором, совместимым с API scikit-learn, например CatBoost, LightGBM, XGBoost и т.д.

Как упоминалось в главе 2, чтобы применить модели машинного обучения к задачам прогнозирования, временной ряд необходимо преобразовать в таблицу, где каждое значение связано с определенным временным окном (известным как задержка), которое ему предшествует. В контексте временных рядов задержка по отношению к временному шагу t определяется как значение ряда на предыдущих временных шагах. Количество задержек, используемых в качестве входных признаков для модели машинного обучения, является важным гиперпараметром, который необходимо тщательно настроить для получения наилучшей производительности модели.

После преобразования данных в новую форму любую регрессионную модель из scikit-learn можно обучить для прогнозирования следующего значения ряда. Во время обучения модели каждая строка считается отдельным экземпляром данных, где значения с задержками $1,2,..p$ считаются предикторами целевого значения временного ряда на временном шаге $p + 1$, как показано на рис. 3.4.

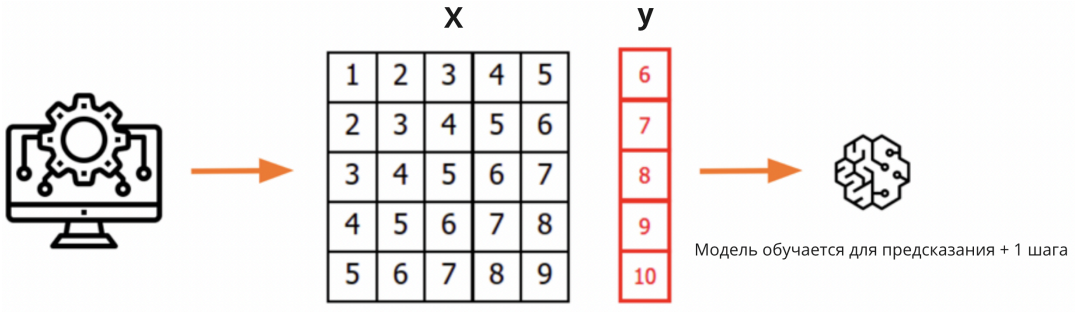

Рис. 3.4. Схема обучения модели машинного обучения с данными временных рядов.

Этот тип преобразования также позволяет включать дополнительные переменные, называемые экзогенными, для прогнозирования временного ряда (рис. 3.5).

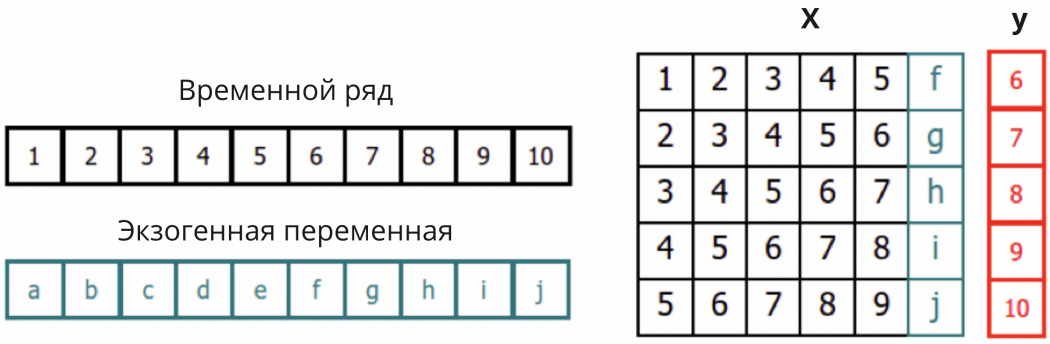

Рис. 3.5. Преобразование временных рядов, включающее экзогенную переменную.

При работе с временными рядами редко требуется прогнозировать только следующий элемент ряда $t + 1$. Вместо этого наиболее распространенная цель – предсказать весь будущий интервал $(t + 1) \ldots (t + n)$ или отдаленный момент времени $(t + n)$. Метод рекурсивного многошагового прогнозирования, реализованный в библиотеке skforecast, позволяет генерировать такой тип предсказания.

Рекурсивное многошаговое прогнозирование предполагает использование предсказанных значений из предыдущих временных шагов в качестве входных данных для прогнозирования значений для последующих временных шагов. Сначала модель прогнозирует на один временной шаг вперед, а затем использует этот прогноз в качестве входных данных для следующего

временного шага, продолжая этот рекурсивный процесс до тех пор, пока не будет достигнут желаемый горизонт прогнозирования.

Поскольку значение t_{n-1} требуется для прогнозирования t_n, а t_{n-1} неизвестно, применяется рекурсивный процесс, в котором каждый новый прогноз основан на предыдущем, как показано на рис. 3.6.

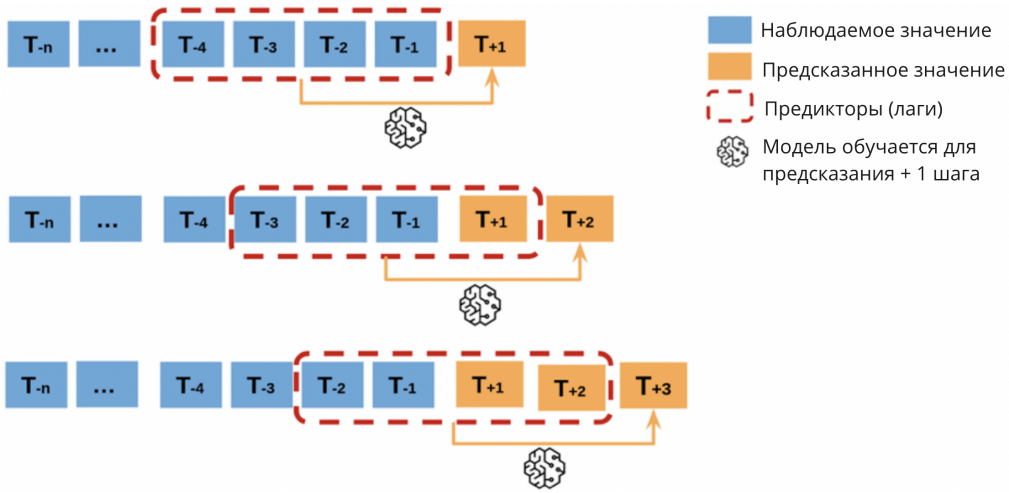

Рис. 3.6. Схема рекурсивного многошагового прогнозирования.

Рекурсивное прогнозирование или рекурсивное многоэтапное прогнозирование и может быть легко реализовано с помощью классов ForecasterAutoreg и ForecasterAutoregCustom из библиотеки skforecast.

3.3.2 Используемые модели

Прогнозирование заболеваемости коронавирусом и вакцинации в данном исследовании реализовано с помощью следующих моделей: ARIMA, экспоненциального сглаживания, k-ближайших соседей, случайного леса, а также XGBoost.

Для идентификации данных временного используется подход, который включает проверку случайности данных и тенденцию в данных временного ряда.

В процессе построения модели ARIMA необходимо провести несколько этапов. На первом этапе необходимо проверить стационарность временного

ряда. Для этого используются тесты Дики-Фуллера или Критерий Адфила. Если временной ряд не является стационарным, то необходимо применить преобразование Бокса-Кокса для приведения временного ряда к стационарному виду.

На следующем этапе необходимо оценить параметры модели ARIMA. Для этого используются такие методы, как анализ графика автокорреляции (ACF) и частной автокорреляции (PACF). Эти методы позволяют оценить параметры модели ARIMA, такие как порядок авторегрессии p, порядок интегрирования d и порядок скользящего среднего q.

Несмотря на то, что этапы обучения модели ARIMA более сложны, ее реализация относительно проста. Для прогнозирования временного ряда с помощью ARIMA можно использовать библиотеку sktime, в которой реализована модель ARIMA, требующая прямого указания порядка модели (p, q, d), а также модель AutoARIMA, не требующая прямого указания порядка модели.

Предсказательная модель на основе экспоненциального сглаживания реализуется на основе класса из библиотеки sktime, который называется ExponentialSmoothing. В настройках по умолчанию используется простое экспоненциальное сглаживание без компонентов тренда и сезонности, но их можно задать с помощью параметров модели. Помимо этого, модель экспоненциального сглаживания, в том числе и тройного экспоненциального сглаживания Хольта-Винтерса, реализуется самостоятельно, используя методику, описанную в разделе 2.1.2.

Для предсказания на основе методов машинного обучения используются модели из библиотеки scikit-learn, такие как KNeighborsRegressor, RandomForestRegressor, XGBRegressor. Данные временных рядов преобразуются в данные поперечного сечения с помощью класса ForecasterAutoreg библиотеки skforecast для многошагового рекурсивного прогнозирования [41].

Модели KNeighborsRegressor, RandomForestRegressor, XGBRegressor поддерживают использования дополнительной экзогенной переменной, например, статистики по вакцинации.

Библиотека skforecast также представляет инструмент для вероятностного прогнозирования временных рядов [41]. В большинстве случаев интерес вызывает конкретный интервал. Чтобы предсказать интервал, в skforecast реализован метод predict_interval. Этот метод внутренне использует predict_bootstrapping для получения матрицы бутстрэпп матрицы и оценивает верхний и нижний квантили для каждого шага, тем самым предоставляя желаемые интервалы прогнозирования. Предсказанные интервалы не зависят от распределения, что означает, что не делается никаких предположений о конкретном распределении данных.

3.3.3. Подбор гиперпараметров модели

Гиперпараметры модели для прогнозирования временных рядов – это параметры, которые необходимо определить для использования модели. Как правило, при настройке гиперпараметров модели сначала выбираются несколько гиперпараметров, которые могут влиять на результат работы модели в рамках конкретной задачи. Затем задается разумный диапазон значений для каждого гиперпараметра. Далее определяется, какая конфигурация является наилучшей, например, в смысле минимизации метрики ошибки.

В большинстве случаев не существует аналитического способа определить, какие гиперпараметры и в какой конфигурации работают лучше всего. В целом нужно поэкспериментировать и посмотреть, что работает с заданными данными.

Одним из популярных методов нахождения «лучшей» конфигурации выбранных гиперпараметров является k-fold кросс-валидация (k-блочная перекрестная проверка).

K-fold кросс-валидация работает путем разделения данных на k блоков примерно одинакового размера. Каждый блок представляет собой подмножество данных с $\frac{N}{k}$ наблюдениями. Затем модель обучается на всех блоках, кроме одного, т.е. $k-1$, и вычисляют ошибку на оставшемся блоке k, который служит валидационной выборкой. Этот процесс повторяется k раз, пока каждый блок не будет использован в качестве валидационного набора. Обычно процесс выбора $k-1$ блоков – случайный, как показано на рис. 3.7.

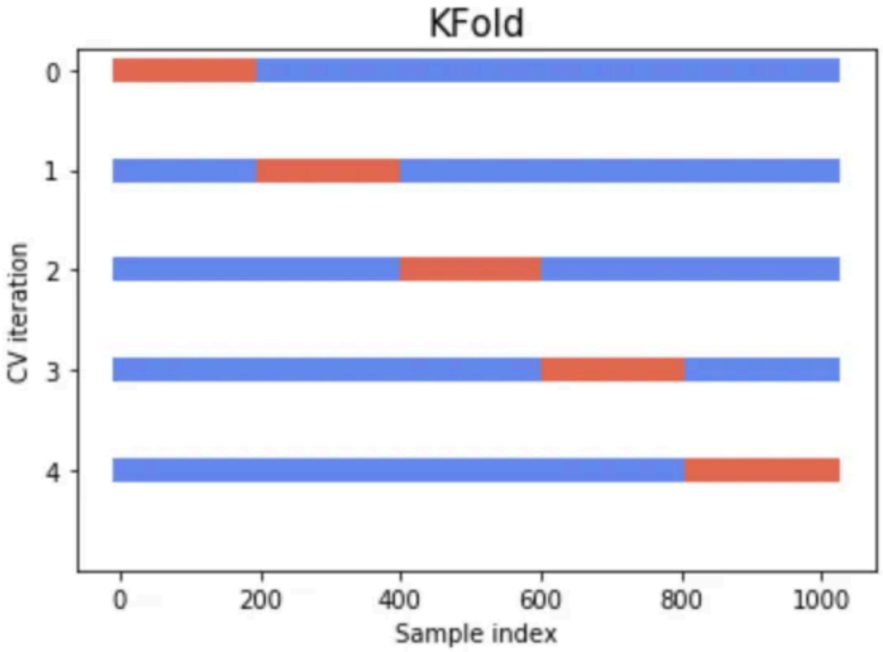

Рис. 3.7. K-fold кросс-валидация

Такой подход нельзя использовать на временных рядах, т.к. не можем выбирать случайные блоки и назначать их либо валидационному набору, либо обучающему набору, чтобы избежать подглядывания в будущее, когда обучаем нашу модель. Между наблюдениями существует временная зависимость, и необходимо сохранять эту связь.

Метод перекрестной проверки модели для прогнозирования временных рядов – это перекрестная проверка на непрерывной основе (cross-validation on a rolling basis). Для этого изначально выбирается небольшая тренировочная выборка, производится прогноз для более поздних точек данных на h шагов вперед, которые служат валидационной выборкой, затем оценивается точность

прогнозируемых точек данных. Затем те же прогнозируемые точки данных включаются как часть следующего обучающего набора данных, и последующие точки данных прогнозируются. При этом размер валидационной выборки – фиксированный.

Процесс кросс-валидации на непрерывной основе показан на рис.3.8.

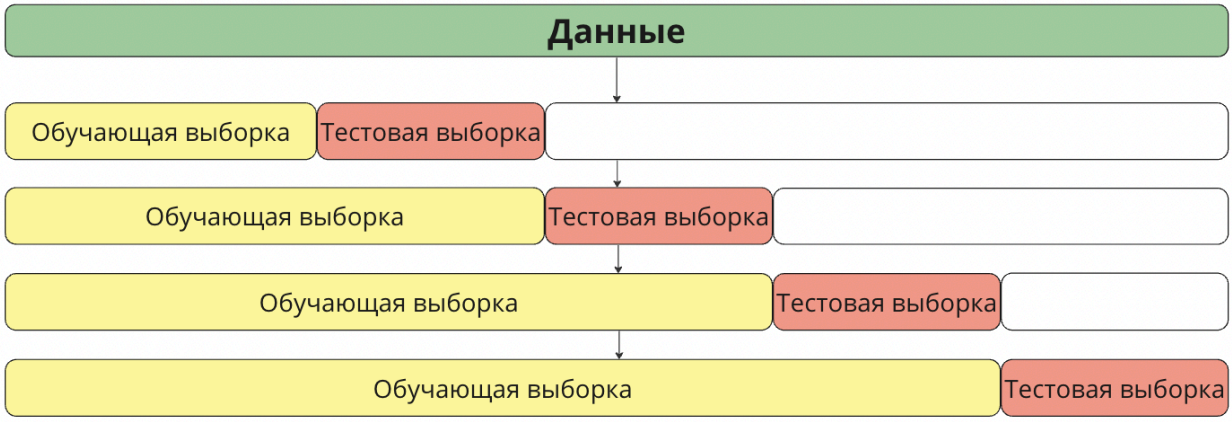

Рис. 3.8. Кросс-валидации на непрерывной основе

Рассмотрим гиперпараметры, которые настаивались для используемых методов прогнозирования.

В ходе построения модели экспоненциального сглаживания задавались некоторые из основных гиперпараметров: коэффициенты модели Хольта-Винтерса (α, β, γ) и длина сезона.

В ARIMA есть несколько гиперпараметров, которые можно настраивать. Несмотря на то, что не все гиперпараметры могут быть настроены, выбор правильных значений для тех, которые настраиваются может повысить качество модели ARIMA. Гиперпараметры, которые настраивались для ARIMA:

- p – порядок авторегрессии, который определяет количество предыдущих значений, которые используются для прогнозирования текущего значения. Подбор этого параметра основан на анализе автокорреляционной функции (ACF) и частичной автокорреляционной функции (PACF) временного ряда.

- d – порядок интегрирования, которые определяет, сколько раз временной ряд должен быть дифференцирован, чтобы сделать его стационарным. Этот параметр определяется на основе теста Дики-Фуллера.
- q – порядок скользящего среднего, который определяет количество предыдущих ошибок, которые используются для прогнозирования текущего значения. Подбор этого параметра также основан на анализе ACF и PACF.
- seasonal_order – параметр, описывающий сезонность временного ряда.

XGBRegressor – это реализация модели градиентного бустинга на основе деревьев решений для задачи регрессии. Для этой модели также можно задавать ряд гиперпараметров, которые влияют на ее качество и производительность. Ниже приведены наиболее важные гиперпараметры, которые использовались при обучении модели XGBRegressor:

- learning_rate – коэффициент скорости обучения, который определяет величину изменения градиента на каждом шаге градиентного бустинга. Маленькое значение может привести к более точным результатам, но увеличить время обучения.
- max_depth – максимальная глубина деревьев решений. Низкое значение может ограничить способность модели к изучению более сложных зависимостей, а слишком большое значение может привести к переобучению модели. Для прогнозирования временных рядов используется глубина деревьев, которая обычно находится в диапазоне от 3 до 10.
- n_estimators – количество деревьев, которые будут использоваться для прогнозирования. Чем больше деревьев, тем более точные прогнозы могут быть получены. Более высокое значение может дать лучшую производительность, но также может привести к переобучению. Для прогнозирования временных рядов используется количество деревьев, которое обычно находится в диапазоне от 10 до 1000.

- booster – частота использования базовой модели (tree или linear) в процессе бустинга. Использовался параметр 'gbtree', который применяется для задачи регрессии на основе деревьев решений.
- objective – функция ошибки, которая минимизируется в процессе обучения модели. Для прогнозирования временных рядов используется функция потерь RMSE (Root Mean Square Error).

Для построения модели случайного леса использовался класс RandomForestRegressor. Существует несколько параметров, которые влияют на качество и производительность модели для прогнозирования временных рядов, в данной работе настраивались следующие:

- n_estimators – количество деревьев в лесу. Чем больше деревьев, тем более точные прогнозы может дать модель.
- criterion – критерий разделения на каждом узле дерева. Обычно в задаче регрессии используются метрика MSE (среднеквадратичное отклонение).
- max_depth - максимальная глубина деревьев решений. Чем больше глубина, тем больше деталей модель может захватить.

KNeighborsRegressor –реализация алгоритма k-ближайших соседей в Python. При обучении модели машинного обучения KNeighborsRegressor использовались следующие гиперпараметры:

- n_neighbors - количество ближайших соседей.
- weights – веса, используемые для учета расстояния до каждого ближайшего соседа. Использовался параметр «distance», при котором вес обратно пропорционален расстоянию до точки.
- p – параметр Минковского для расчета расстояния между точками. При $p = 2$ используется расстояние Евклида.

Выбор оптимальных гиперпараметров для моделей производился с помощью оптимизации на основе кросс-валидации на непрерывной основе. Все гиперпараметры устанавливаются таким образом, чтобы минимизировать

ошибку прогнозирования. Для алгоритма случайного леса и градиентного бустинга оптимизация проводилась для гиперпараметров: max_depth и n_estimators, а для k-ближайших соседей – для n_neighbors. В случае экспоненциального сглаживания ошибка минимизировалась на основе гиперпараметров α, β, γ.

3.3.4. Оценка точности работы моделей

Производительность прогнозной модели принято оцениваеть с помощью следующих метрик:

- Среднеквадратическая ошибка или MSE;
- Квадратный корень среднеквадратическая ошибка или RMSE;
- Средняя абсолютная ошибка или MAE;
- Средняя абсолютная ошибка в процентах или MAPE;
- Медианная абсолютная ошибка в процентах или MedAPE.

Среднеквадратическая ошибка определяется как разница между истинным значением временного ряда и спрогнозированным значением, возведенная в квадрат.MSE определяется следующим выражением:

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

где y_i – i-ое истинное значение в наборе данных;

$\hat{y}_i$ - i-ое прогнозируемое значение.

Возведение в квадрат также имеет эффект увеличения больших ошибок. Это приводит к тому, что модели больше «штрафуют» за большие ошибки, чем за маленькие.

RMSE является расширением среднеквадратичной ошибки и описывается формулой:

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

Средняя абсолютная ошибка в отличие от MSE и RMSE не придает большее или меньшее значение различным типам ошибок, вместо этого итоговая ошибка увеличивается линейно с увеличением ошибки на i-ом шаге.

MAE рассчитывается как среднее значение абсолютных значений ошибок следующим образом:

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

Средняя абсолютная ошибка в процентах широко используется, потому что ее легко интерпретировать и объяснить, и выражается как:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{y_i} * 100\%$$

Медианная абсолютная ошибка так же, как и MAPE, является наиболее интуитивно понятной и описывается формулой ниже. При этом $MedAPE$ более устойчива к выбросам.

$$MedAPE = median\,(|y_1 - \hat{y}_1|, \dots, |y_n - \hat{y}_n|)$$

ГЛАВА 4. АНАЛИЗ РЕЗУЛЬТАТОВ

В этой главе будут приведены результаты, полученные в ходе исследования, в том числе оценка влияния вакцинации на заболеваемость и смертность от коронавирусной инфекции, а также будут представлены результаты работы моделей и оценка ошибки полученных прогнозов.

4.1. Анализ влияния вакцинации на заболеваемость и смертность

Анализ влияния вакцинации на заболеваемость и смертность проведен по методике, описанной в 3 главе.

Оценка влияния одного показателя на другой производится с помощью определения коэффициента корреляции. Значения коэффициента корреляции должны находиться в диапазоне от -1 до 1. При этом коэффициент корреляции, равный -1, показывает идеальную отрицательную корреляцию, коэффициент корреляции, равный 1, показывает идеальную положительную корреляцию.

Рассмотрим полученные результаты оценки корреляции статистики по вакцинации и статистики по заболеваемости на примере штата Аляска.

Исходный временной ряд по вакцинации и заболеваемости показан на рис. 4.1.

Рис. 4.1. Статистика по вакцинации и заболеваемости в штате Аляска.

График полученной зависимости коэффициента кросс-корреляции от лагов в неделях временного ряда по вакцинации показан на рис 4.2.

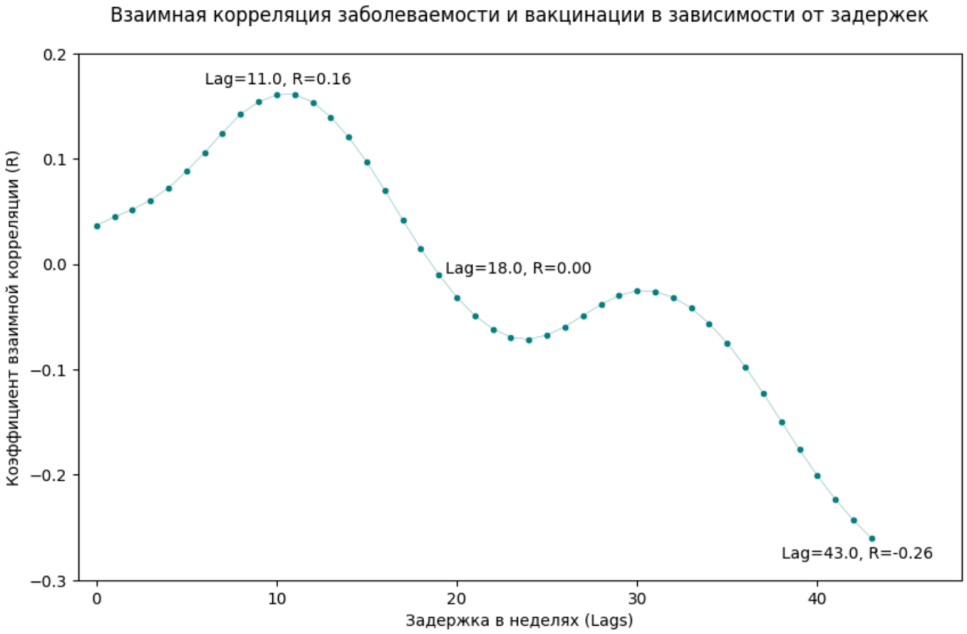

Рис. 4.2. Влияние вакцинации на заболеваемость.

Коэффициент корреляции невысокий, что говорит о слабом влиянии. При этом с увеличением количества задержек коэффициент корреляции снижается и постепенно становится отрицательным. Это говорит о том, что вакцинация может влиять на заболеваемость после некоторого периода времени, но слабо, т.к. коэффициент корреляции низкий. Также стоит отметить, коэффициент корреляции переходит в диапазон отрицательных значений после 18 недели.

Оценку корреляции статистики по вакцинации и по смертности также рассмотрим на примере штата Аляска.

Исходный временной ряд по вакцинации и смертности показан на рис. 4.3-4.4.

Рис. 4.3. Статистика по смертности в штате Аляска.

Рис. 4.4. Статистика по вакцинации в штате Аляска.

График полученной зависимости коэффициента кросс-корреляции от лагов в неделях временного ряда по вакцинации показан на рис 4.5.

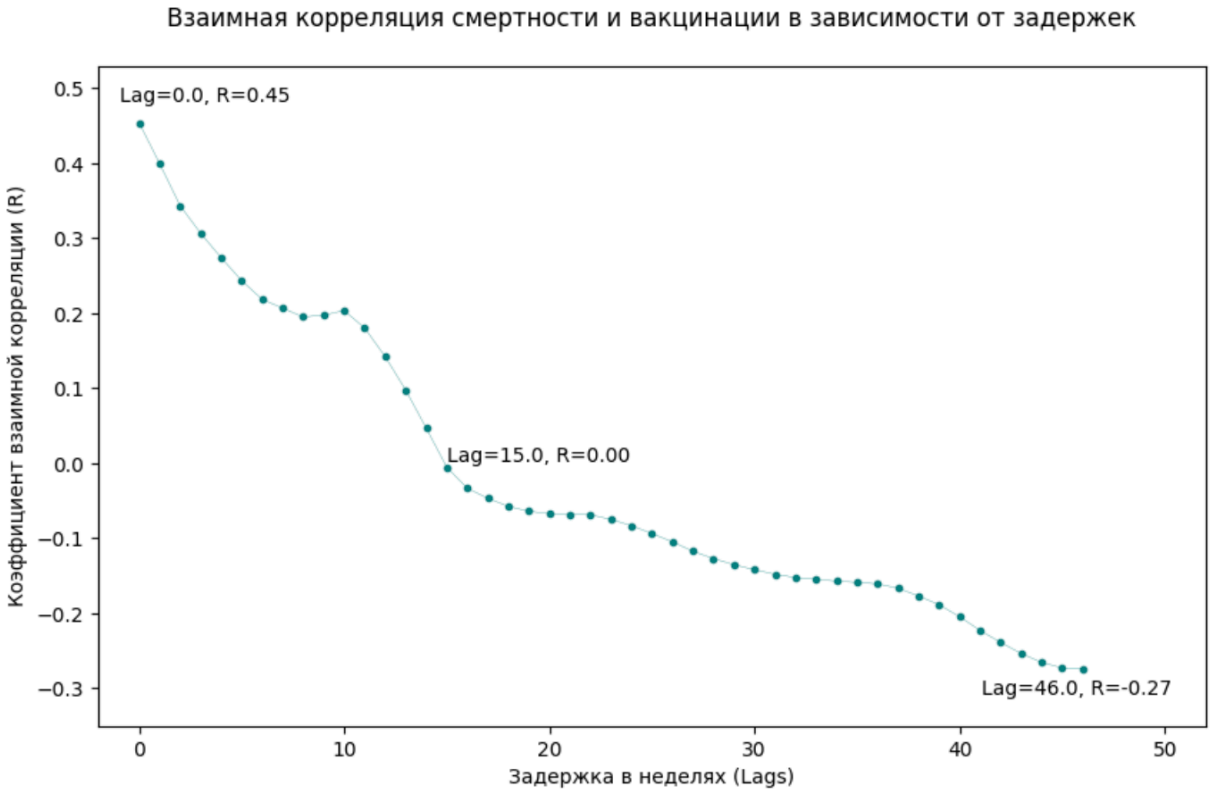

Рис. 4.5. Влияние вакцинации на смертность.

Коэффициент корреляции так же, как и в предыдущем случае, невысокий и постепенно снижается, переходя в диапазон отрицательных значений. Т.е. вакцинация также слабо влияет на снижение смертность с течением времени. При этом коэффициент корреляции переходит в диапазон отрицательных значений после 15 недели.

Результаты влияния вакцинации на заболеваемость и смертность для остальных штатов аналогичны, показанным выше. Несмотря на то, что влияние несильное, статистика по вакцинации использовалась в качестве предиктора для моделей машинного обучения для повышения точности прогноза.

4.2. Визуализация полученных результатов прогнозирования

Прежде чем приступить к прогнозу была проведена предварительная обработка данных, а также временные ряды были проверены на стационарность

с помощью теста Дики-Фуллера. Гипотеза о нестационарности временных рядов была отвернута, т.к. p-значение меньше уровня значимости $\alpha = 5\%$.

Для прогнозирования еженедельного суммарного количества заражений Covid-19 были применены два статистических метода: ARIMA и экспоненциальное сглаживание и три модели машинного обучения: k-ближайших соседей, случайный лес и градиентный бустинг. При этом для прогнозирования на основе методов машинного обучения использовалась статистика по вакцинации в качестве предиктора. Также метод экспоненциального сглаживания был реализован самостоятельно, без использования готовой модели из библиотек Python.

Вероятностный прогноз был построен для всех моделей, кроме экспоненциального сглаживания.

Рассмотрим полученные результаты прогнозирования на примере штата Аляска и Калифорния. Итоговая статистика ошибок прогнозов для всех штатов будет представлена в следующем разделе.

Результат точечного прогнозирования еженедельного суммарного количества заражений Covid-19 для Аляски и Калифорнии с использованием модели тройного экспоненциального слаживания представлен на рис. 4.6-4.7 соответственно.

Рис. 4.6. Прогноз заболеваемости Covid-19 в Аляске с помощью алгоритма тройного экспоненциального сглаживания.

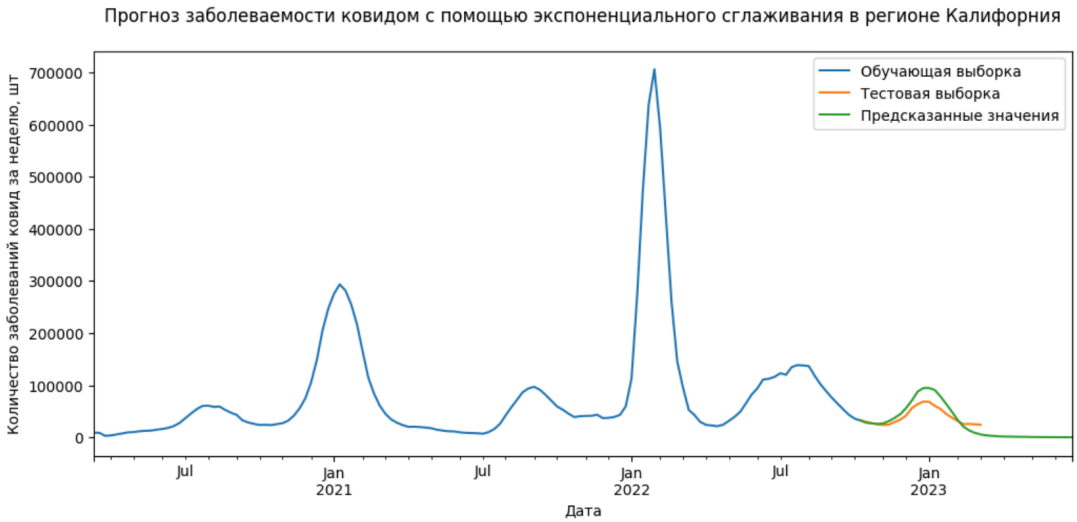

Рис. 4.7. Прогноз заболеваемости Covid-19 в Калифорнии с помощью алгоритма тройного экспоненциального сглаживания.

Точность модели оценивается с помощью нескольких ошибок, представленных в таблице 4.1.

Таблица 4.1. Ошибки модели экспоненциального сглаживания.

Ошибка/Штат	MAPE	MedAPE	MAE	MSE	RMSE
Аляска	85%	66%	4213	25271901	5027
Калифорния	30%	30%	12563	245714075	15675

Результат точечного и вероятностного прогнозирования еженедельного суммарного количества заражений Covid-19 с использованием модели ARIMA для Аляски и Калифорнии представлен на рис. 4.8-4.9. Зеленым цветом на рисунке выделен спрогнозированный интервал, в пределах которого будет находиться суммарное количества заражений Covid-19 за конкретную неделю с вероятностью 90%.

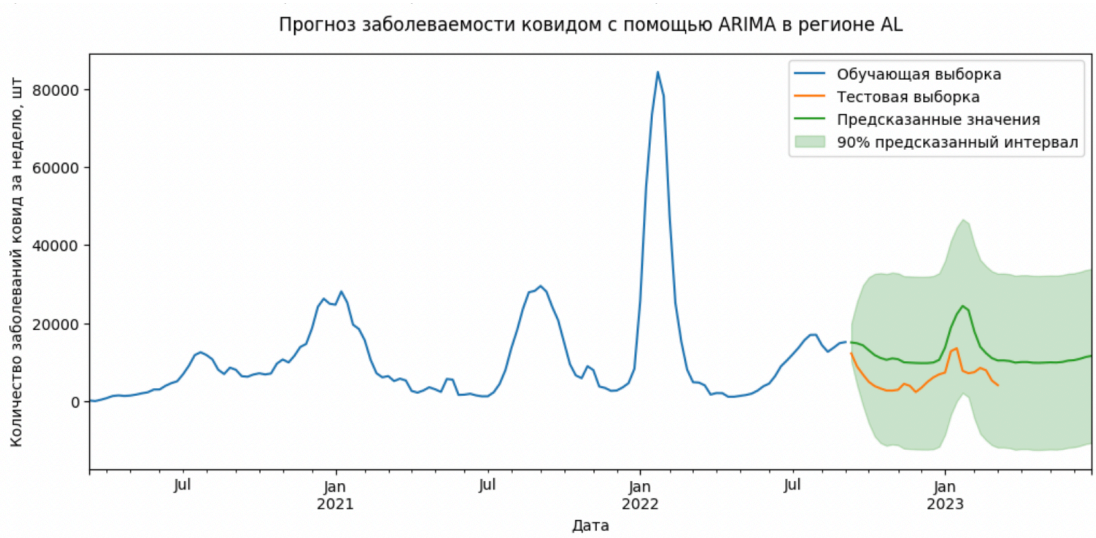

Рис. 4.7. Прогноз заболеваемости Covid-19 в Аляске с помощью алгоритма ARIMA.

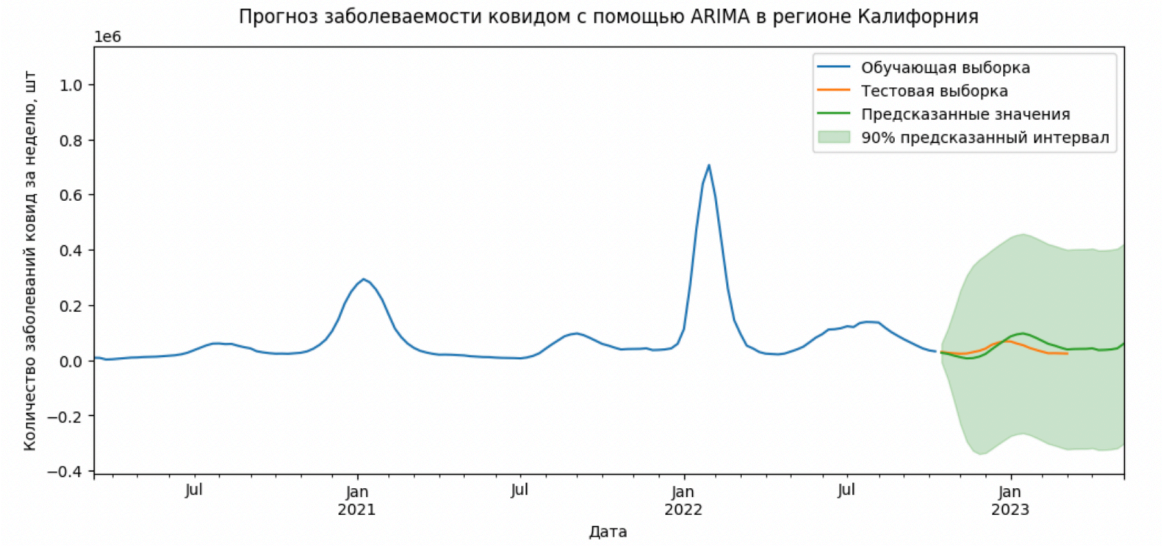

Рис. 4.8. Прогноз заболеваемости Covid-19 в Калифорнии с помощью алгоритма ARIMA.

Метрики ошибок для модели ARIMA представлены в таблице 4.2.

Таблица 4.2. Ошибки модели ARIMA.

Ошибка/Штат	MAPE	MedAPE	MAE	MSE	RMSE
Аляска	132%	125%	6387	54268245	7366
Калифорния	62%	60%	21609	646639804	25429

Результат точечного и вероятностного прогнозирования еженедельного суммарного количества заражений Covid-19 с использованием модели k-ближайших соседей для Аляски и Калифорнии представлен на рис. 4.9-4.10. Зеленым цветом на рисунке выделен спрогнозированный интервал, в пределах которого будет находиться суммарное количества заражений Covid-19 за конкретную неделю с вероятностью 90%.

Рис. 4.9. Прогноз заболеваемости Covid-19 в Аляске с помощью алгоритма k-ближайших соседей.

Рис. 4.10. Прогноз заболеваемости Covid-19 в Калифорнии с помощью алгоритма k-ближайших соседей.

Метрики ошибок для модели k-ближайших соседей представлены в таблице 4.3.

Таблица 4.3. Ошибки модели k-ближайших соседей.

Ошибка/Штат	MAPE	MedAPE	MAE	MSE	RMSE
Аляска	55%	43%	2711	8320829	2884
Калифорния	75%	74%	24490	670052863	25885

На рис. 4.11-4.12 аналогично предыдущим случаем представлены точечный и вероятностный прогнозы для модели случайного леса в регионах Аляска и Калифорния.

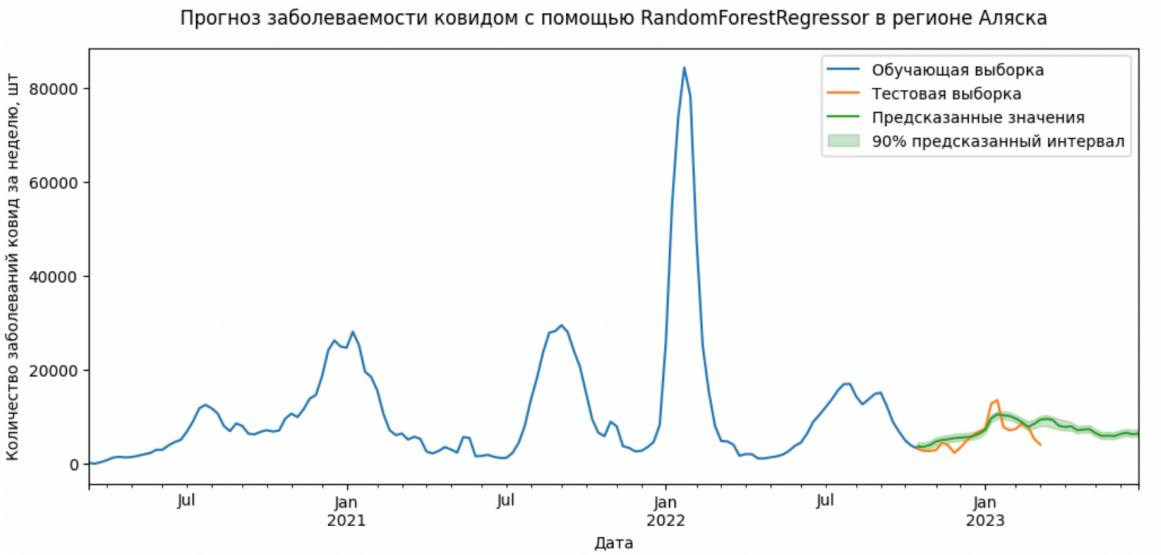

Рис. 4.11. Прогноз заболеваемости Covid-19 в Аляске с помощью алгоритма случайного леса.

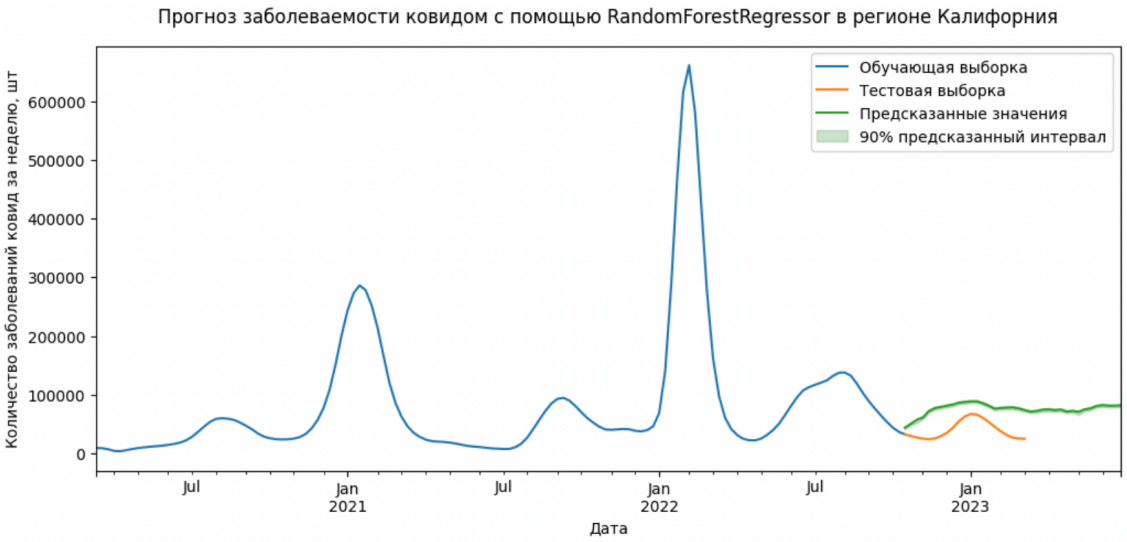

Рис. 4.12. Прогноз заболеваемости Covid-19 в Калифорнии с помощью алгоритма случайного леса.

Аналогично, метрики ошибок для модели случайного леса представлены в таблице 4.4.

Таблица 4.4. Ошибки модели случайного леса.

Ошибка/Штат	MAPE	MedAPE	MAE	MSE	RMSE
Аляска	36%	29%	1711	4781447	2186
Калифорния	110%	104%	35933	1439458755	37940

На рис. 4.13-4.14 представлены точечный и вероятностный прогнозы для алгоритма градиентного бустинга в штате Аляска и Калифорния соответственно.

Рис. 4.13. Прогноз заболеваемости Covid-19 в Аляске с помощью алгоритма градиентного бустинга.

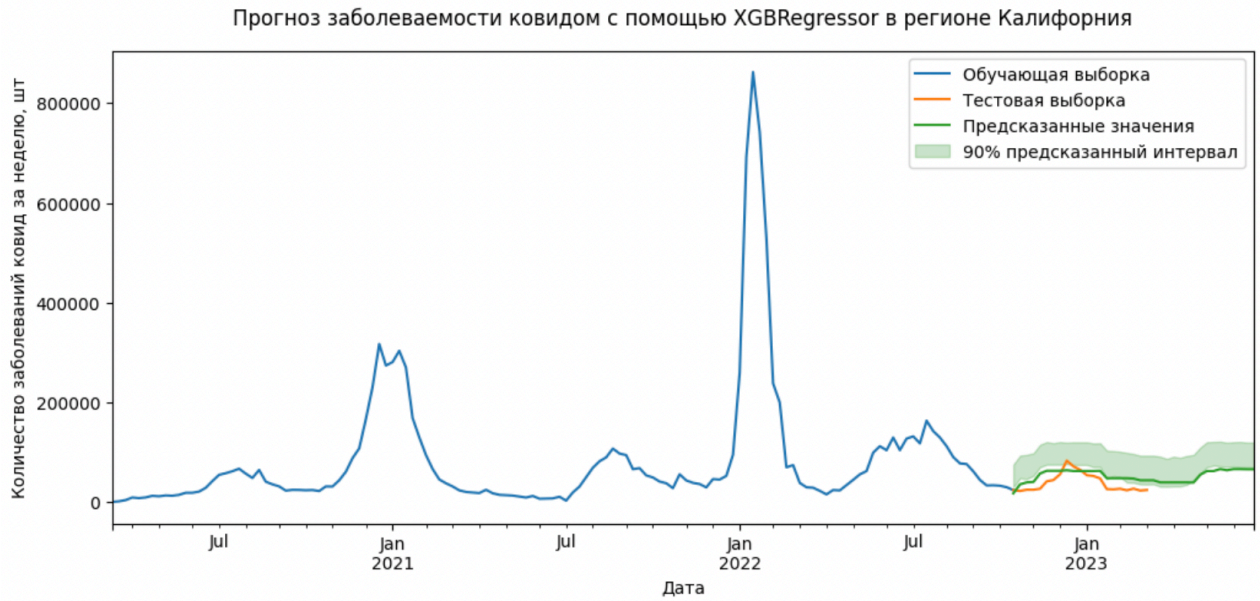

Рис. 4.14. Прогноз заболеваемости Covid-19 в Калифорнии с помощью алгоритма градиентного бустинга.

Оценка точности прогноза для модели градиентного бустинга представлены в таблице 4.5.

Таблица 4.5. Ошибки модели градиентного бустинга.

Ошибка/Штат	MAPE	MedAPE	MAE	MSE	RMSE
Аляска	31%	22%	1854	8904027	2983
Калифорния	54%	59%	16249	310779979	17628

Итоговые результаты оценки точности работы моделей с помощью метрики MAPE для штата Аляска и Калифорния с использованием статистики по вакцинации и без представлены в таблице 4.6-4.7 соответственно.

Таблица 4.6. Ошибка моделей для штата Аляска.

Модель	Экспоненциальное сглаживание	ARIMA	k-ближайших соседей	Случайный лес	XGBoost
MAPE	85%	132%	61%	40%	34%
MAPE с использованием статистики по вакцинации	-	-	55%	36%	31%

Таблица 4.7. Ошибка моделей для штата Калифорния.

Модель	Экспоненциальное сглаживание	ARIMA	k-ближайших соседей	Случайный лес	XGBoost
MAPE	30%	62%	80%	115%	56%
MAPE с использованием статистики по вакцинации	-	-	75%	110%	54%

Из рис.4.6-4.14 можно сделать вывод, что все модели достаточно хорошо подстраиваются под характер временного ряда по заболеваемости, несмотря на высокие ошибки (таблицы 4.6-4.7).

4.3. Сравнение моделей

Прогнозирование заболеваемости выполнялось итеративно для каждого штата отдельно. В таблице 4.8 приведены средняя ошибка по всем штатам для каждой модели.

Таблица 4.8. Средняя ошибка моделей по всем штатам

Модель	Экспоненциальное сглаживание	ARIMA	k-ближайших соседей	Случайный лес	XGBoost
MAPE	102%	154%	90%	87%	58%
MAPE с использованием статистики по вакцинации	-	-	79%	80%	53%

Исходя из полученных результатов, статистические модели хуже справляются с прогнозированием, чем модели машинного обучения. При этом самая высокая точность наблюдается у модели градиентного бустинга, которая также обладает высокой скоростью работы, поэтому является пригодной для работы в промышленности. Случайный лес и k-ближайших соседей также неплохо справляются с прогнозированием, но k-ближайших соседей отличается высокой ресурсоемкостью и низкой скоростью работы.

ЗАКЛЮЧЕНИЕ

В данной работе оценивается возможность применения вероятностного моделирования для прогнозирования развития пандемии на примере исторических данных о заболеваемости Covid-19 в каждом штате США. Для этого строится прогноз с использованием статистических моделей и алгоритмов машинного обучения. Выбор наилучшего алгоритма производится путем сравнения предсказанных значений временного ряда с исходным временным рядом на основе ошибки MAPE и наилучшего соответствия характеру временного ряда. Для повышения точности прогноза анализируется возможность использования дополнительной информации по вакцинации.

В ходе исследования были выполнены поставленные задачи:

- Анализ корреляции вакцинации и заболеваемости;
- Прогнозирование заболеваемости на основе исторических данных;
- Вероятностное моделирование развития пандемии в зависимости от количества вакцинированных людей с использованием алгоритмов машинного обучения и статистических методов;
- Оценка точности работы моделей и сравнение моделей;

В результате работы был создан готовый инструмент на языке Python, который производит предварительную обработку данных и делает прогноз в будущее. Ошибка моделей MAPE варьируется от 30% до 140% на примере одного штата и от 50% до 200%, но при этом предсказанные интервалы достаточно неплохо описывают характер временного ряда, его тренд и сезонность. Стоит отметить, что, если в данных есть сильные выбросы и выраженные сезонные или циклические компоненты, MAPE может недооценивать или завышать ошибки прогноза. Это связано с тем, что MAPE рассматривает процентное отклонение прогнозов от фактических значений, а описательная способность процентного отношения может быть недостаточной для описания сложных циклических процессов.

Результаты исследования также показывают, что использование дополнительной информации по вакцинации позволяет повысить точность прогноза на 2-6%, несмотря на слабое влияние вакцинации на заболеваемость.

Таким образом, полученные в данной работе прогнозные модели развития пандемии COVID-19 могут оказаться полезными для принятия защитных мер и разработки плана действий в период пандемий и эпидемий.

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1. Box G. E. P. et al. Time series analysis: forecasting and control. – John Wiley & Sons, 2015.
2. Brown R. G. Statistical forecasting for inventory control. – 1959.
3. Kotu V., Deshpande B. Data science: concepts and practice. – Morgan Kaufmann, 2018.
4. Montgomery D. C., Johnson L. A., Gardiner J. S. Forecasting and time series analysis. – McGraw-Hill Companies, 1990.
5. Azari A. Bitcoin price prediction: An ARIMA approach //arXiv preprint arXiv:1904.05315. – 2019.
6. Benvenuto D. et al. Application of the ARIMA model on the COVID-2019 epidemic dataset //Data in brief. – 2020. – С. 105340.
7. Breiman L. Random forests //Machine learning. – 2001. – С. 5-32.
8. Ceylan Z. Estimation of COVID-19 prevalence in Italy, Spain, and France //Science of The Total Environment. – 2020. – С. 138817.
9. Chen X. et al. Impact of vaccination on the COVID-19 pandemic in US states //Scientific reports. – 2022. – №. 1. – С. 1554.
10. De La Vega E., Flores J. J., Graff M. K-nearest-neighbor by differential evolution for time series forecasting //Nature-Inspired Computation and Machine Learning: 13th Mexican International Conference on Artificial Intelligence, MICAI 2014, Tuxtla Gutiérrez, Mexico, November 16-22, 2014. Proceedings, Part II 13. – Springer International Publishing, 2014. – С. 50-60.
11. Djakaria I., Saleh S. E. Covid-19 forecast using Holt-Winters exponential smoothing //Journal of Physics: Conference Series. – IOP Publishing, 2021. – №. 1. – С. 012033.
12. Dudek G. Short-Term Load Forecasting using Random Forests. – 2011.

13. Fischer A. et al. Statistical learning for wind power: A modeling and stability study towards forecasting //Wind Energy. – 2017. – №. 12. – С. 2037-2047.
14. Friedman J. H. Greedy function approximation: a gradient boosting machine //Annals of statistics. – 2001. – С. 1189-1232.
15. Goehry B. et al. Random forests for time series. – 2021.
16. Holt C. C. Forecasting trends and seasonals by exponentially weighted moving averages //ONR Memorandum. – 1957. – №. 52. – С. 5-10.
17. Huang C. et al. Correlation between vaccine coverage and the COVID-19 pandemic throughout the world: Based on real-world data //Journal of Medical Virology. – 2022. – №. 5. – С. 2181-2187.
18. Kane M. J. et al. Comparison of ARIMA and Random Forest time series models for prediction of avian influenza H5N1 outbreaks //BMC bioinformatics. – 2014. – №. 1. – С. 1-9.
19. Kumar M., Anand M. An application of time series ARIMA forecasting model for predicting sugarcane production in India //Studies in Business and Economics. – 2014. – №. 1. – С. 81-94.
20. Kumar Y. et al. Machine Learning and Deep Learning Based Time Series Prediction and Forecasting of Ten Nations' COVID-19 Pandemic //SN Computer Science. – 2022. – №. 1. – С. 91.
21. Lahouar A., Slama J. B. H. Random forests model for one day ahead load forecasting //IREC2015 The Sixth International Renewable Energy Congress. – IEEE, 2015. – С. 1-6.
22. Li X. et al. Probabilistic solar irradiance forecasting based on XGBoost //Energy Reports. – 2022. – С. 1087-1095.
23. Liu L. et al. Predicting the incidence of hand, foot and mouth disease in Sichuan province, China using the ARIMA model //Epidemiology & Infection. – 2016. – №. 1. – С. 144-151.

24. Luo J. et al. Time series prediction of COVID-19 transmission in America using LSTM and XGBoost algorithms //Results in Physics. – 2021. – С. 104462.
25. Martínez F. et al. Dealing with seasonality by narrowing the training set in time series forecasting with kNN //Expert systems with applications. – 2018. – С. 38-48.
26. Martínez F. et al. Time Series Forecasting with KNN in R: the tsfknn Package //R J. – 2019. – №. 2. – С. 229.
27. Moftakhar L., Mozhgan S., Safe M. S. Exponentially increasing trend of infected patients with COVID-19 in Iran: a comparison of neural network and ARIMA forecasting models //Iranian Journal of Public Health. – 2020. – №. Suppl 1. – С. 92.
28. Moon J. et al. Hybrid short-term load forecasting scheme using random forest and multilayer perceptron //Energies. – 2018. – №. 12. – С. 3283.
29. Ostertagova E., Ostertag O. Forecasting using simple exponential smoothing method //Acta Electrotechnica et Informatica. – 2012. – №. 3. – С. 62.
30. Oyewola D. O. et al. Predicting COVID-19 cases in South Korea with all K-edited nearest neighbors noise filter and machine learning techniques //Information. – 2021. – №. 12. – С. 528.
31. Poleneni V., Rao J. K., Hidayathulla S. A. COVID-19 prediction using ARIMA model //2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence). – IEEE, 2021. – С. 860-865.
32. Rügamer D. et al. Probabilistic time series forecasts with autoregressive transformation models //Statistics and Computing. – 2023. – №. 2. – С. 37.
33. Sidqi F., Sumitra I. D. Forecasting product selling using single exponential smoothing and double exponential smoothing methods //IOP Conference Series: Materials Science and Engineering. – IOP Publishing, 2019. – №. 3. – С. 032031.

34. Tajmouati S. et al. Applying k-nearest neighbors to time series forecasting: two new approaches //arXiv preprint arXiv:2103.14200. – 2021
35. Talavera-Llames R. L. et al. A nearest neighbours-based algorithm for big time series data forecasting //Hybrid Artificial Intelligent Systems: 11th International Conference, HAIS 2016, Seville, Spain, April 18-20, 2016, Proceedings 11. – Springer International Publishing, 2016. – С. 174-185.
36. Tyralis H., Papacharalampous G. A review of probabilistic forecasting and prediction with machine learning //arXiv preprint arXiv:2209.08307. – 2022.
37. Van der Meer D. W. et al. Probabilistic forecasting of electricity consumption, photovoltaic power generation and net demand of an individual building using Gaussian Processes //Applied energy. – 2018. – С. 195-207.
38. Wan C. et al. Probabilistic forecasting of photovoltaic generation: An efficient statistical approach //IEEE Transactions on Power Systems. – 2016. – №. 3. – С. 2471-2472.
39. Wang Y., Shen Z., Jiang Y. Comparison of ARIMA and GM (1, 1) models for prediction of hepatitis B in China //PloS one. – 2018. – №. 9. – С. e0201987.
40. Zhang L. et al. Time series forecast of sales volume based on XGBoost //Journal of Physics: Conference Series. – IOP Publishing, 2021. – №. 1. – С. 012067.
41. Documentation. Skforecast. – 2022. – URL: https://skforecast.org/0.7.0/introduction-forecasting/introduction-forecasting.html. – (дата обращения: 15.05.2023).
42. Documentation. Sktime. – 2022. – URL: https://www.sktime.net/en/latest/users.html. – (дата обращения: 15.05.2023).
43. Johns Hopkins Coronavirus Resource Center. — URL: https://coronavirus.jhu.edu/map.html. – (дата обращения: 20.01.2023).