МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий
(наименование института полностью)

Кафедра «Прикладная математика и информатика»
(наименование)

01.03.02 Прикладная математика и информатика
(код и наименование направления подготовки / специальности)

Компьютерные технологии и математическое моделирование
(направленность (профиль)/специализация)

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)

на тему «Разработка алгоритма учета и анализа динамики заболеваемости сотрудников компании»

Обучающийся А.С. Шуляев
(Инициалы Фамилия) (личная подпись)

Руководитель М.А. Тренина
(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Консультант Е.В. Косс
(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Тольятти 2022

Аннотация

Тема выполненной бакалаврской работы «Разработка алгоритма учета и анализа динамики заболеваемости сотрудников компании».

Работа выполнена студентом Тольяттинского государственного университета, института математики, физики и информационных технологий, группы ПМИб-1802б, Шуляевым Алексеем Сергеевичем. Объектом исследования являются методы решения задачи прогнозирования временных рядов. Целью работы является разработка алгоритма для анализа и учета данных. Для того чтобы реализовать данную идею необходимо

1) Исследовать предметную область.

2) Реализовать интерфейс и выбранный метод на языке программирования Python.

3) Проверить интерфейс и работоспособность разработанного метода при помощи различных метрик.

Отчет состоит из введения, трех разделов и заключения.

В первой части представлены основные теоретические данные, определения и выбранный метод прогнозирования. Во второй части работы происходит описание проектирование модели прогнозирования временных рядов и создание интерфейса для приложения учета данных. В третьей части проводится тестирование реализованного метода и интерфейса.

Данная бакалаврская работа состоит из 51 страницы, содержит 19 иллюстраций, 8 таблиц, список используемой литературы состоит из 25 источников.

Подводя итоги, хотелось бы подчеркнуть, что данная работа актуальна не только в решении проблемы с изучением эпидемиологической ситуации внутри определенной компании, но и в том, что она может быть применена в других сферах.

Abstract

The title of the graduation work is "Development of an algorithm for recording and analyzing morbidity rate of employees."

The work was done by a student of Togliatti State University, Institute of Mathematics, Physics and Information Technologies, group PMIb-1802b, Shulyaev Aleksey Sergeevich. The object of research is the methods for solving the problem of time series forecasting. The aim of the work is to develop an algorithm for analyzing and monitoring data. In order to implement this idea, it is necessary to:

1) Investigate the subject area of the research.

2) Create the interface and selected method using the Python programming language.

3) Check the interface capabilities and efficiency of the developed method by using various metrics.

The senior paper consists of an introduction, three chapters and a conclusion.

The first part presents the main theoretical information, definitions and the chosen forecasting method. The second part of the work describes a time series forecasting model and the creation of an interface for a data monitoring application. In the third part, testing of the created method and interface is conducted.

This senior paper consists of 51 pages, contains 19 illustrations, 8 tables, the list of references consists of 25 sources.

In conclusion, I would like to emphasize that this work is relevant to solving the problem of studying the epidemiological situation within a particular company, as well as it can be used in other areas.

Содержание

Введение

Любая компания заинтересована в том, чтобы сотрудники не болели. Если кто-то из работников все-таки уходит на больничный, то начинают сдвигаться сроки реализации проекта, доход падает и за этим следует еще множество факторов, влияющих на стабильную работу организации. Они решаемы, но от этого компании не легче. Несмотря на всем ныне известный Covid-19, нужно помнить, что по мимо него существуют и другие заболевания, которым подвергаются люди. Для управления ситуацией вокруг данной проблемы, нужно понимать, что вообще происходит.

Прогнозирование в сфере здравоохранения внутри предприятий необходимо для того, чтобы верно принимать превентивные меры и избегать нежелательные исходы для компании и её сотрудников.

Это подводит нас к изучению конкретных методов прогнозирования в целом, ведь от правильности метода и оценки её достоверности зависит точность будущего прогноза.

Актуальность бакалаврской работы обусловлена в решении проблемы с изучением эпидемиологической ситуации внутри определенной компании.

Объектом исследования являются методы решения задачи прогнозирования временных рядов.

Цель работы: разработка алгоритма для анализа и учета данных.

Для достижения цели данной бакалаврской работы потребуется выполнить следующие шаги:

1) Исследовать предметную область;

2) Рассмотреть различные модели прогнозирования и найти подходящую для поставленного случая;

3) Реализовать интерфейс приложения и выбранный метод при помощи Python и облачного сервиса Google Colab;

4) Проверить интерфейс и работоспособность разработанного метода при помощи различных метрик.

1 Обзор предметной области

1.1 Основные понятия временных рядов

Временной ряд – это последовательность вещественных значений, которая определенна во времени [9]. Главным фактором является равно удаленность отсчетов времени друг от друга. Благодаря этому мы можем говорить о равной дискретизации значений временного ряда и можем оценить тенденцию дальнейшего развития данной характеристики. У каждого процесса есть начало и конец.

В качестве примера Временного ряда можно привести:

– количество заболевших внутри компании за месяц;

– учет товаров в магазине за день;

– объем осадков за год.

Пример временного ряда также изображен на рисунке 1.

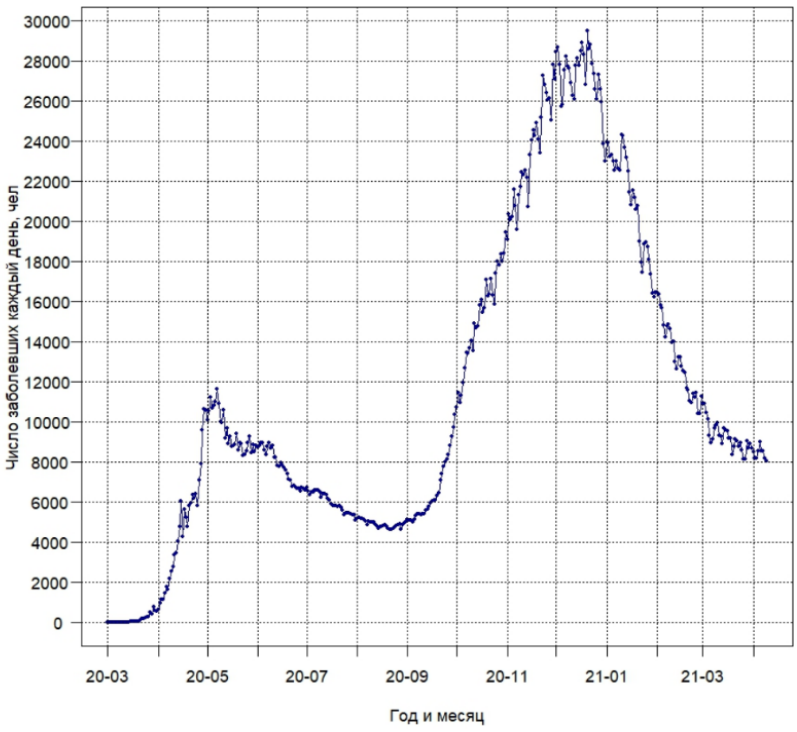

Рисунок 1 – Динамика числа заболевших Covid-19 в России

Формально, временной ряд может быть записан в следующем виде:

Последовательность вещественных значений $y \in R^{N+1}$ измеренных через одинаковые промежутки времени $\Delta t \in R$:

$$y_i = y(t_i), \quad (1)$$
$$t_i = t_0 + i * \Delta t,$$
$$i = 0,1, \dots, N,$$

где t_i – время замеров;

i – отсчеты по времени;

t_0 – характерное время начала процесса.

Временные ряды могут быть:

– одномерными;

– многомерными (неравномерными).

Одномерный временной ряд – это единичное наблюдаемое значение действительное во времени. Пример одномерного ряда представлен на рисунке 2.

Рисунок 2 – Запасы нефти на месторождении Северного Кавказа

Многомерный временной ряд – это временной ряд, в котором значения параметров ряда находятся в зависимости от значений предшествующих показателей этого же и других рядов. Пример представлен на рисунке 3.

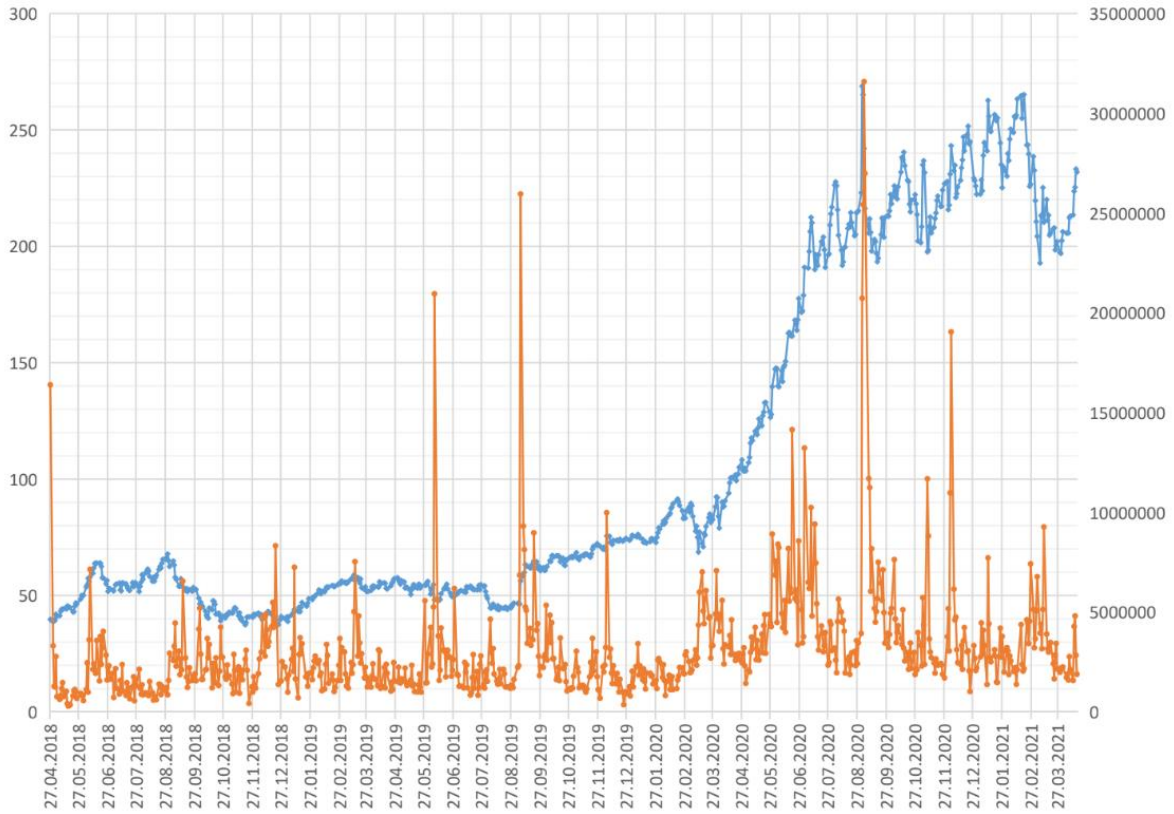

Рисунок 3 – Данные о цене акций компании

1.2 Задачи временных рядов

Идентификация временных рядов – это процесс опознания их статистических свойств, которое поможет их проще обрабатывать в зависимости от задачи.

Прогнозированием временных рядов называют создание модели для анализа определенного события, значения которого будут изменяться в будущем, при этом используя данные которые происходили с ним в прошлом.

В качестве примера можно взять прогноз на стоимость акций компании в связи с её предыдущей деятельностью.

Уровнем временного ряда называют среднее значение временного ряда.

Остатки временного ряда – это параметр который отображает разницу между фактическими и предсказанными значениями. Благодаря этим данным появляется возможность оценить определенный метод прогнозирования. Метод с остатками, обладающий максимальной точностью прогнозов должен обладать следующими свойствами:

– остатки метода не должны коррелировать между собой. В них не должно оставаться полезной для построения более точного прогноза;

– прогноз не должен быть смещенным, в том случае, если его остатки равны нулевому среднему значению.

Одним из ключевых аспектов в построении модели, считается глубокий анализ изначального временного ряда. Поиск идеальной модели не возможен, потому, что на данный момент не существует абсолютно универсальной модели, которая создавала бы идеальный прогноз для всех типов рядов.

Классификацией временных рядов называют разбиение его по конкретным характеристикам. Примером можно считать классификацию временных рядов, которая зависит их вида. Ряды делятся на моментные и интервальные.

Моментный ряд – это временной ряд, хранящий внутри себя характеристики исследуемого явления в различные моменты времени. В качестве примера моментных рядов можно считать поголовье скота в фермерских хозяйствах на 1 января или 20 августа за несколько лет, количество определенного материала на складе в определенный месяц, численность новых граждан страны в начале года, периода и т.д.

В качестве примера моментного ряда могут служить следующие данные представленные в таблице 1.

Таблица 1 – Численность населения РФ (на 1 января).

Год	1992	1996	2000	2004	2007	2012	2016	2020
Показатель	148,5	148,2	146,6	144,1	142,8	143,2	144,3	144,1

Интервальный (периодический) временной ряд – последовательность, в которой уровень исследуемого явления считается результатом, созданным или накопленным за определенное количество времени. Примером могут являться размер имеющегося товара по месяцам года, отдельные периоды времени отработанных человеком дней (месяцев, кварталов, лет, и т.п.) и т.д.

Примером такого ряда могут служить данные о добычи нефти в Российской Федерации (Таблица 2).

Таблица 2 – Добыча нефти в Российской Федерации.

Год	2014	2015	2016	2017	2018	2019	2020	2021
млн. т.	526,8	534,08	547,5	546,7	555,84	560,2	505,6	524,05

Кластеризация – это задача, которая позволяет разделить на классы большую совокупность временных рядов по метрике.

Поиск аномалий – это выискивание внутри временного ряда значений которые не соответствуют нормам этого ряда. В различных областях деятельности аномалии могут представать перед нами в различных видах:

- точечные аномалии – это несоответствия, которые появляются в определенных местах;
- групповые аномалии – это отклонения в норме предстающие перед нами в виде определенного скопления точек, но в том случае, когда мы берем одну из них отдельно, то аномальной она считаться не будет;

– аномалии контекста – это системные отклонения, которые никак не относятся к значениям внутри ряда. Примером таких аномалий является минусовая температура в тропическом регионе летом.

Эта процедура очень важна. Для вывода более точного прогноза, необходимо устранять аномалии внутри рядов. Исследование этих явлений считается в некоторых ситуациях необходимым. Примером таких ситуаций могут быть оповещения операционной системы о возможном скором переходе устройства в аварийный режим.

1.3 Статистические свойства временных рядов

В классических задачах анализа данных и машинного обучения главной особенностью временных рядов считается независимость наблюдений подвыборки.

При прогнозировании временных рядов, как и в задачах моделирования языка, мы надеемся на то, что значения ряда в прошлом содержат содержательную информацию о поведении ряда в будущем с сохранением тенденции на определённом интервале [23].

Данная особенность при обработке и анализе временных рядов позволяет нам не только строить прогнозы для рядов значений во времени, но и решать другие задачи обработки временных рядов.

Из чего состоят временные ряды на качественном уровне:

$$y(t_i) = f(T, S, C, E, t_i). \quad (2)$$

Цикличность C(t) – несистематическое изменения уровня ряда с переменным периодом. Её часто можно встретить в экономических, экологических, физических процессах.

Сезонность S(t) – это плавные или резкие изменения уровней ряда в определенном фрагменте времени, постоянно меняющееся в течение

заданного периода. Пример временного ряда с сезонностью изображен на рисунке 4.

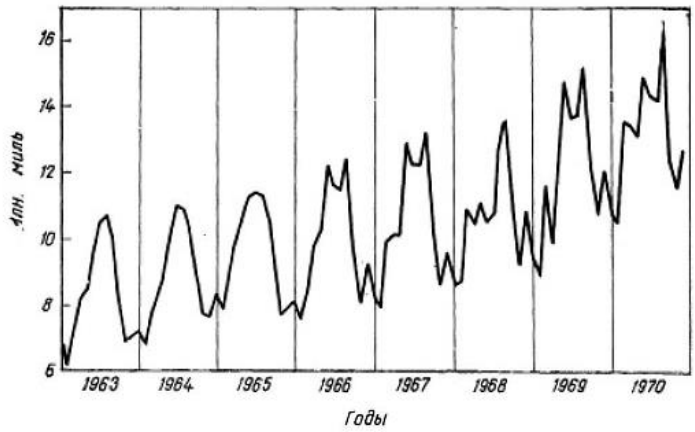

Рисунок 4 – Временной ряд со свойством сезонности

Тренд T(t) – плавное изменение уровня ряда во времени. (рисунок 5).

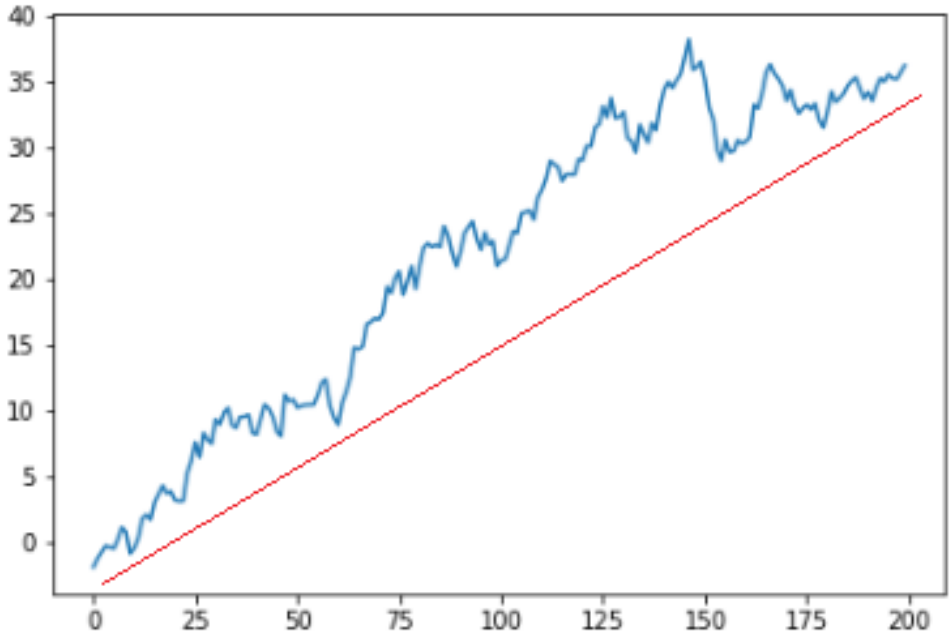

Рисунок 5 – Временной ряд со свойством тренда

Ошибка E(t) – непрогнозируемая случайная компонента ряда, которая не всегда может влиять на временной ряд [16]. Благодаря ей можно создать предиктивный прогноз (рисунок 6).

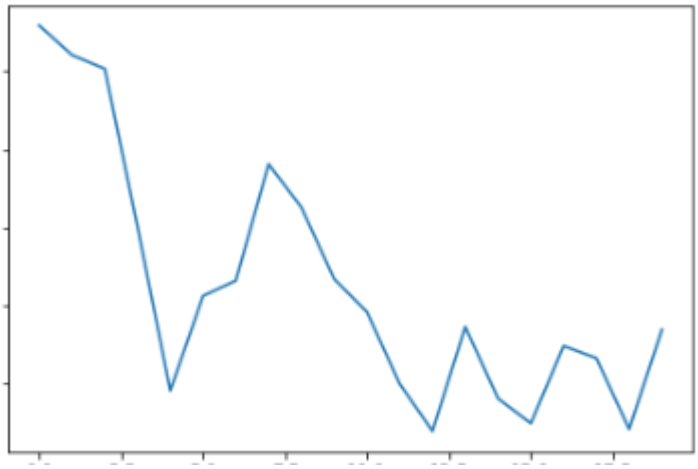

Рисунок 6 – Демонстрация случайной составляющей во временном ряду

Автокорреляция – это корреляционная зависимость значений временного ряда, которые сменяют друг друга. Появляется в том случае, когда соседствующие между собой значения взаимосвязаны.

Число периодов, по которым рассчитывается называется лагом.

Лаг – это количество моментов, по которым принято рассчитывать коэффициент автокорреляции. Лаговый оператор B сначала берет значение элемента временного ряда и уменьшает его на единицу времени. Если лаговый оператор используется снова, то значение сдвигается еще на несколько временных единиц. Расчет шагов лага происходит по следующей формуле:

$$\begin{gathered} By_t = y_{t-1}, \\ B(By_t) = B^2 y_t = y_{t-2}, \\ B^p y_t = y_{t-p}. \end{gathered} \tag{3}$$

Обычно временные ряды описывают при помощи следующих критериев:

– математическое ожидание – это средний параметр произвольного размера, измерения которого стремятся к бесконечности;

– дисперсия – это случайная очередность параметров произвольного размера по отношению к математическому ожиданию;

– автокорреляционная функция – это очередность коэффициентов автокорреляции с лагами со случайными значениями не меньше единицы.

Временные ряды как правило делят на стационарные и нестационарные.

Для анализа временных рядов используется стационарность в широком смысле [19]. Временной ряд, который будет иметь следующую совокупность свойств, будет называется стационарным в широком смысле:

– математическое ожидание ряда не меняется во времени. Иными словами, у ряда отсутствует тренд:

$$M(t_1) = M(t_2) = \cdots = M(t_N), \quad (4)$$

– ковариация смещённого на k отсчётов назад ряда не зависит от времени:

$$Cov(y_t, y_{t-k}) = \tau_k. \quad (5)$$

Пример стационарного ряда представлен на рисунке 7.

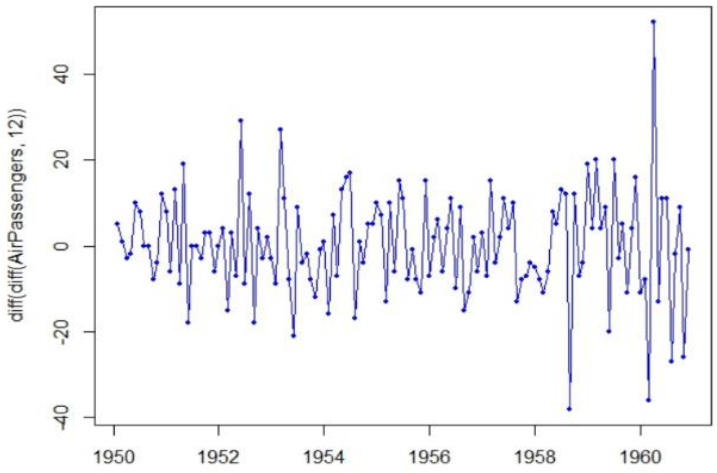

Рисунок 7 – Пример стационарного ряда

Главным отличием нестационарные временные ряда от стационарного является наличие преднамеренные элементы. Такими могут быть:

– ряды с трендом;

– ряды с сезонностью;

– комбинированные ряды (ряды, имеющие тренд и сезонность одновременно).

Пример нестационарного ряда представлен на рисунке 8.

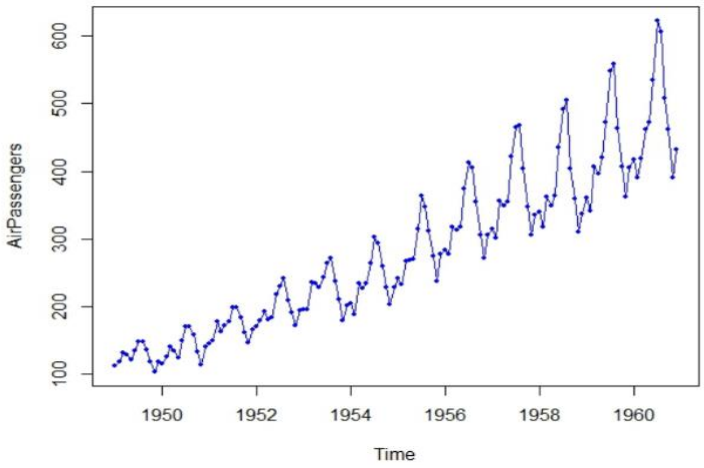

Рисунок 8 – Пример нестационарного ряда

У нестационарного ряда есть возможность превращения в стационарный. Для того, нужно воспроизвести следующий алгоритм:

– если у нестационарного временного ряда обнаружится возможность экспоненциального роста, то для него используют простое логарифмирование или логарифмирование цепных индексов:

$$y_t^* = \ln(y_t), \tag{6}$$

$$y_t^* = \ln\left(\frac{y_t}{y_{t-1}}\right) = \ln y_t - \ln y_{t-1}, \tag{7}$$

– следующим шагом является вычисление роста исследуемого временного ряда при помощи следующей функции:

$$y_t^* = \frac{y_t - y_{t-1}}{y_{t-1}} = \frac{y_t}{y_{t-1}} - 1, \tag{8}$$

– «чтобы преобразовать стационарные ряды применяют еще один популярный метод, интегрирование или взятие конечных разностей различных порядков временного ряда» [7, 8]. Интегрирование первого порядка можно представить в виде следующего уравнения:

$$\Delta y_t = y_t - y_{t-1}. \tag{9}$$

Интегрирование для порядка d можно представить при помощи следующего уравнения:

$$\Delta^d y_t = \Delta^{d-1} y_t - \Delta^{d-1} y_{t-1}. \tag{10}$$

Для того, чтобы выполнить анализ временного ряда необходимы различные методы аналитики для выборки из него необходимых элементов.

1.4 Тест Дикки-Фуллера

При помощи этого теста проверяют является ли ряд стационарным или нет. Он проверяет ряд на наличие единичного корня в авторегрессии на один шаг назад. Если говорить конкретно, то проверяется значение коэффициента α в авторегрессионном уравнении первого порядка:

$$y_t = \alpha * y_{t-1} + \varepsilon_t, \quad (11)$$

где y_t– является временным рядом,

ε_t– ошибка.

В том случае, когда значение параметра α приравнивается к единице, процесс имеет единичный корень, а это обозначает что временной ряд не является стационарным.

Если |α| <1, то ряд стационарный. Тест Дикки-Фуллера рассчитывает p-статистику, в случае p <0.05 гипотеза о стационарности ряда не отвергается [13].

Из-за его простоты тест работает не очень хорошо. Существует довольно много улучшенных тестов таких как:

- расширенный тест Дикки-Фуллера;
- Kwiatkowski–Phillips–Schmidt–Shin (KPSS).

1.5 Построение модели временного ряда

Приступим к построению модели временного ряда. Для этого используется следующий алгоритм:

- происходит сглаживание изначального временного ряда. Это необходимо для облегчения нахождения общей тенденции переменчивых значений временного ряда при помощи изменения изначальных данных на

предполагаемые данные, которые как правило гораздо меньше подвержены хаотичным изменениям. Для того чтобы выравнивание было возможным, как правило используют метод простой скользящей средней, при использовании данного метода первоначальные данные уровней ряда преобразуют в средние арифметические значения на заданных интервалах;

– поиск сезонной (S) компоненты для аддитивной модели начинается с вычисления среднего значения для всех значений ряда при помощи с показателя абсолютного отклонения $S \rightarrow S_{\Delta i}$. Основным требованием для этого, является необходимость приравнивать к нулю общее значение сезонных компонент. Поиск сезонной компоненты для мультипликационных моделей необходимо представлять в виде общего урезанного значения ряда. Поэтому применяют показатель индекса сезонности $S \rightarrow I_{si}$. По итогу у сезонных компонент общее среднее значение должно ровняться единице;

– следующим шагом происходит вычитание сезонной компоненты из изначального ряда. После этого на выход мы подается ряд с отлаженными параметрами уровня;

– далее аналитически выравниваются уровни ошибочной и трендовой компонент и после этого происходит поиск значений T компоненты. Выбор многочлена происходит благодаря анализу уже сглаженного графика уровней временного ряда и конкретизации вида трендовой модели;

– проводится поиск трендовых и ошибочных компонент;

– вычисление абсолютных и относительных ошибок при помощи вырезания из временного ряда сезонной и трендовой компонент.

После этого необходимо удостоверится в наличии автокорреляции. «Для нахождения автокорреляции как правило применяют критерий Дарбина-Уотсона» [1]. Изначально считают, что значения остатков E_t для всех

временных значений t не являются зависимыми от значений остатков для других t. Для проверки данных гипотез предоставляется следующая формула:

$$d = \frac{\sum_{t=2}^{n}(\varepsilon_i - \varepsilon_{i-1})^2}{\sum_{t=1}^{n}\varepsilon_t^2}, \quad (12)$$

при $\varepsilon_t = y_t - \hat{y}_t$.

В случае, когда задается новое значение Дарбина-Уотсона равному нулю можно сказать, что значения ошибок не имеют автокорреляции. В таком случае можно считать, что исходные уровни ряда можно заменить данными значениями ошибок.

1.6 Разбор методов прогнозирования

Методы прогнозирования рядов подразделяются на несколько видов:

– модели от времени (экстраполяция на время);
– модели авторегрессии;
– гибридные модели.

Наибольшее распространение в сфере автоматического прогнозирования получили методы построения авторегрессионных моделей ввиду простоты построения обучающего множества и отсутствия привязки к отдельным предикторам [6].

1.6.1 Модели от времени

Модели прогнозирования по времени имеют некоторые общие подходы к построению.

Дискретный ряд данных y_i рассматривается как некоторый непрерывный во времени процесс, зависящий от времени, а также нескольких других составляющих, таких как тренд, сезонность, случайные ошибки и возможно экзогенные переменные.

За счёт исследования временного ряда на сезонность имеется возможность за счёт техник разделения ряда на тренд и сезонность оценить будущую тенденцию развития процесса.

1.6.2 Модели авторегрессии

Пусть имеется исходный временной ряд $y(t_i)$ с моментами отсчётов $i = 0, 1, ..., N;$ тогда моделью авторегрессии для прогнозирования данного временного ряда является соотношение:

$$\hat{y}(t_{i+d}) = f(y(t_{i-1}), y(t_{i-2}), y(t_{i-3}), \dots, y(t_{i-tw}), \theta), \quad (13)$$

где d - количество отсчётов вперёд, которое мы хотим пропустить;

tw - окно авторегрессионной зависимости,

θ - множество параметров модели f.

Сэмплирование для авторегрессии производится скользящим окном с шириной авторегрессионной зависимости. В данном случае пространство признаков для предсказываемой величины будет иметь размерность равную ширине окна tw.

1.7 Модели экстраполяции на основе кривых роста

Из-за того, что универсальной модели для построения всех типов рядов не существует, нам необходимо провести анализ изначального временного ряда и выбрать идеальную для нашего случая модель. Мой выбор пал на модели экстраполяции на основе кривых роста.

В этих моделях как правило самым частым используемым качеством является тренд, ведь во время применения этой характеристики возможно создавать, как долгосрочные, так и краткосрочные прогнозы. «Обычно берут в пример сразу несколько моделей тренда.» [14, 15].

Пусть y_t –значение временного ряда в момент времени t, а X_t – момент времени ($X_t = t = (0, 1, 2, \dots, n)$.

Моделью линейного тренда называют самую легкую модель, которую можно использовать для построения прогноза.

Данная модель имеет вид:

$$\widehat{y_t} = b_0 + b_1 * X_t, \quad (14)$$

где b_i – характеристические данные модели.

Эта модель представлена как прямая линия и как правило, её используют чтобы расписать происходящий во времени однотипный момент процесс. [28].

Моделью полиномиального тренда называют один из самых легких в применении нелинейных трендов, который представлен в следующем виде:

$$\widehat{y_t} = b_0 + b_1 * X_t + b_2 * X_t^2 + \cdots + b_p * X_t^p, \quad (15)$$

где p – порядок полинома.

Модель полиномиального тренда, в определенных случаях, может стать моделью линейного тренда, например, если значение *p=1.*

Коэффициенты многочленов приобретают конкретное воплощение в минимальных степенях, которое зависит от составляющей временного ряда.

Примером этого может быть ряд в котором b_0 – уровень ряда при значении $t = 0$, b_1 – быстрота увеличения роста, b_2 – ускорение, показатель которого равен половине скорости значения показателя, а b_3 – перемена уровня ускорения.

Метод наименьших квадратов может быть применен чтобы найти оценки коэффициентов моделей. Основная идея этого метода заключается в том, что благодаря нему становится возможным нахождение параметров, при которых общая сумма квадратов отклонений расчетных значений уровней от фактических была бы минимальной и оценки параметров после этого могут быть вычислены при помощи минимизации выражения:

$$\sum_{t=1}^{n}(y_t - \widehat{y_t})^2 \to min, \tag{16}$$

где n – размер временного ряда;

y_t – фактические показатели уровней временного ряда;

$\widehat{y_t}$ – расчетные данные.

Систему нормальных уравнений для определения параметров модели получают на основании то, что функция достигает минимальное значение в критических точках, а именно в точках, в которых частные производные этой функции равны нулю. Таким образом, можно систему нормальных уравнений записать в следующем виде:

$$\begin{cases} b_0 n + b_1 \sum t + b_2 \sum t^2 + \cdots + b_p \sum t^p = \sum y_t \\ b_0 \sum t + b_1 \sum t^2 + b_2 \sum t^3 + \cdots + b_p \sum t^{p+1} = \sum y_t * t \\ b_0 \sum t^{p-1} + b_1 \sum t^p + b_2 \sum t^{p+1} + \cdots + b_p \sum t^{2p-1} = \sum y_t * t^{p-1} \\ \cdots\cdots\cdots\cdots \\ b_0 \sum t^p + b_1 \sum t^{p+1} + b_2 \sum t^{p+2} + \cdots + b_p \sum t^{2p} = \sum y_t * t^p \end{cases} \tag{17}$$

Гиперболическую модель тренда применяют, чтобы расписывать заканчивающиеся процессы, которые имеют асимптоту при значении *y=0.* Эта модель записывается в следующем виде:

$$\hat{y}_t = b_0 + \frac{b_1}{X_t}. \tag{18}$$

В таких случаях система нормальных уравнений, оценивающая параметр b, будет выглядеть следующим образом:

$$\begin{cases} nb_0 + b_1 \sum \frac{1}{X_t} = \sum y_t \\ b_0 \sum \frac{1}{X_t} + b_1 \sum \frac{1}{X_t^2} = \sum \frac{1}{X_t} * y_t \end{cases}. \quad (19)$$

Бывают довольно повторяющиеся случаи, когда значения экстраполяции прогнозируемых значений ряда происходит вычисление при помощи интервального прогноза. Из-за этого появляется необходимость в поиске доверительного интервала прогноза. Это происходит при помощи следующей формулы:

$$\hat{y}_{n+l} \pm t_\alpha S_y \sqrt{\frac{n+1}{n} + \frac{(t_l - \bar{t})^2}{\sum (t - \bar{t})^2}}, \quad (20)$$

где S_y – среднее квадратическое отклонение фактических данных от расчетных.

$$S_y = \sqrt{\frac{\sum (y_t - \hat{y}_t)^2}{n - k}}, \quad (21)$$

где t_α – показатель *t*-статистики Стьюдента;

k – Количество параметров кривой подвергнутых оцениванию;

t – Номер позиции уровня ряда;

$\bar{t}$ – номер позиции уровня, который находится в центре ряда ($\bar{t} = \frac{(n+1)}{2}$).

Эти модели довольно часто используются в прогнозировании довольно не простых процессов, не зависимо от их простоты. С их использованием прогнозы получаются довольно надежными, в основном когда модель тренда адекватно разъясняет главную тенденцию развития.

1.8 Адаптивные методы прогнозирования

У выбранных нами адаптивных методов прогнозирования есть одна особенность – в данном методе есть возможность учета аномалий исследуемых процессов и адаптация к ним [7, 11].

В адаптивных методах основной идеей является способность создать корректирующую модель с учетом прогноза, который был совершен на предыдущем шагу [3, 21]. Характеристика сглаживания отображает скорость осознания моделью колебаний динамики временного ряда. Из-за этого адаптивные методы довольно часто используют для быстрого или краткосрочного прогноза. Обычно выделяют следующие три модели:

– однопараметрическая модель Брауна;
– двухпараметрическая модель Хольта;
– трехпараметрическая модель Хольта-Винтерса.

Модель Брауна – данная модель сглаживает временной ряд при поддержке, взвешенной скользящей средней. Главной особенностью скользящей средней считается подчинение всех весов экспоненциальному закону, который можно описать следующим образом: «Более ранним наблюдениям присваиваются меньшие веса, чем поздним» [5]. Данную модель иногда называют экспоненциальным сглаживанием.

Сглаживание, как правило, считают методом, который дает возможность выбора некоторых данных, которые хранят внутри себя полезную информацию и шумы. Визуально сглаживание представлено на рисунке 9.

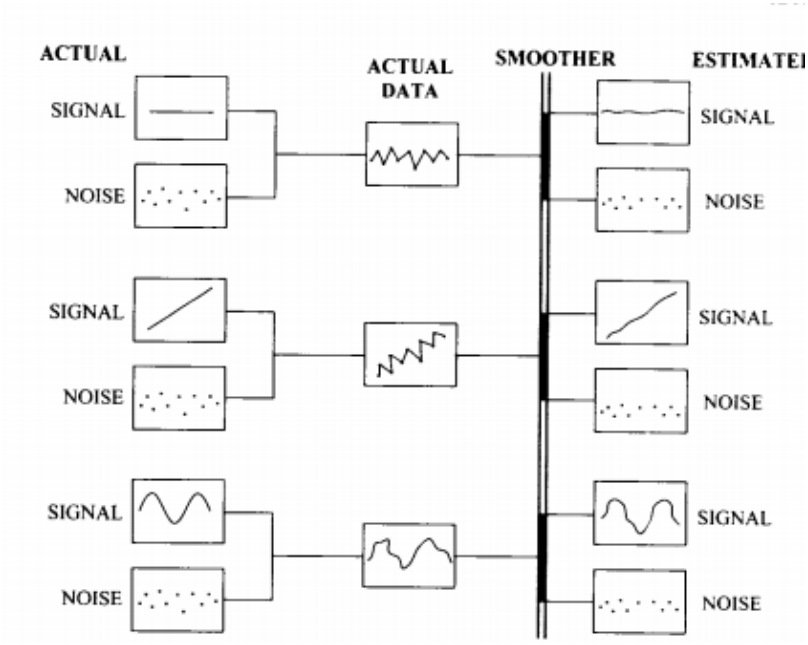

Рисунок 9 – Описание процесса сглаживания

В модели Брауна возможно регулярное обновление модели, которое происходит благодаря предоставлению новых данных. Отобразить прогноз возможно с помощью следующего выражения:

$$\hat{y}_{t+1} = \alpha y_t + (1-\alpha)\hat{y}_t, \quad (22)$$

где $\hat{y}_{t+1}$ – значение прогноза;

α – характеристика сглаживания ($0 < \alpha < 1$);

y_t – фактическое значение ряда в фрагмент времени *t*;

$\hat{y}_t$ – предыдущий прогноз.

После раскрытия скобок в выражении 22, мы получим следующую формулу:

$$\hat{y}_{t+1} = \hat{y}_t + \alpha(y_t - \hat{y}_t). \quad (23)$$

Модель Брауна по сути является старым прогнозом с небольшим утверждением α , которое умножают на значение ошибки предыдущего прогноза.

Первым параметром $\hat{y}_0$ может стать изначальное значение временного ряда или же среднее арифметическое уравнений ряда [13, 17].

Выполнение поиска этого значения для сглаживания α , считается совершенно другой задачей, которая находится в зависимости у изначального временного ряда. В том случае, когда необходимо, чтобы значения прогноза являлись постоянными и происходило сглаживание случайных отклонений, значение α должно быть минимальным. В противном случае, если параметр α приобретает наибольшее значение, то модель будет моментально реагировать на волнения динамики временного ряда. Для особенных случаев поиска значения в модели Брауна существует следующая формула:

$$\alpha = \frac{2}{m+1}, \tag{24}$$

где m – значения наблюдений, которые содержатся внутри интервала сглаживания.

В том случае если временной ряд не имеет большой длины, то значение m можно приравнять к длине ряда n.

Чтобы найти идеальный параметр сглаживания, применяется критерий минимального значения среднего квадрата ошибок. Первым делом выберем значение α, которое будет приближено к нулю. Далее с отступом примерно в 0.1, принимаются новые значения (0.1, 0.2, …, 0.9). У всех этих параметров находится средний квадрат ошибок и ищут наименьшее значение ошибки для α. Чтобы построить прогноз на несколько шагов вперед, используется следующее выражение:

$$\hat{y}_{t+l} = \hat{y}_{t+l-1}, \quad (25)$$

где l – количество шагов.

Из-за того, что модель Брауна не считает линейный тренд появилась необходимость в её улучшении. Так появилась модель двойного экспоненциального сглаживания – модель Хольта.

Эта модель в отличии от её предшественника может считать локальный линейный тренд и в том случае если изначальный временной ряд стремиться к росту, то помимо оценки текущего уровня ряда требуется оценить наклон [12]. Для формирования точного прогноза в данной модели существуют следующие три уравнения, работающие сообща.

Первым уравнением является базовое уравнение, которое применяет сглаживание к изначальному временному ряду:

$$a_t = \alpha_1 y_t + (1 - \alpha_1)(a_{t-1} + b_{t-1}). \quad (26)$$

Вторым уравнением является тренд, представленный в виде разницы двух сглаженных значений, изменяющийся во времени:

$$b_t = \alpha_2(a_t - a_{t-1}) + (1 - \alpha_2)b_{t-1}. \quad (27)$$

Третье уравнение требуется для проведения прогноза на l количество шагов, при этом объединяя внутри себя параметры остальных двух уравнений:

$$\hat{y}_{t+l} = a_t + lb_t. \quad (28)$$

Поиск характеристик сглаживания α_1 и α_2 происходит также, как и в модели Брауна. Поиск первичных значений a_0 и b_0 происходит в уравнении линейной регрессии:

$$\hat{y}_{t+1} = c_0 + c_1 t, \quad (29)$$

при этом $c_0 = a_0, c_1 = b_0$.

Недостатком модели Хольта является тот факт, что она считает тренд, но не сезонность. Для того чтобы считать, как сезонность, так и тренд, применяют метод тройного экспоненциального сглаживания, другими словами, модель Хольта-Винтерса.

Данная модель является модернизацией модели Хольта. Она считает тренд, сезонность и значение временного ряда. Модель Хольта-Винтерса создает прогноз опираясь на эти три характеристики, и описывается в четырех уравнениях: [20, 22].

– уравнение сглаживания ряда:

$$L_t = \alpha \frac{y_t}{S_{t-s}} + (1-\alpha)(L_{t-1} + T_{t-1}), \quad (30)$$

– уравнение оценки тренда:

$$T_t = \beta(L_t - L_{t-1}) + (1-\beta)T_{t-1}, \quad (31)$$

– уравнение оценки сезонности:

$$S_t = \gamma \frac{y_t}{L_t} + (1-\gamma)S_{t-s}, \quad (32)$$

– уравнения прогноза на l шагов вперед:

$$\hat{y}_{t+l} = (L_t + lT_t)S_{t-s+l}. \quad (33)$$

при этом s – период сезонности (4 для квартальных данных, 12 для годовых). α, β, γ – параметры сглаживания.

Во время построения модели нам нужно найти параметры сглаживания α, β, γ. Их поиск возможен благодаря минимизации средней квадратической ошибки. Первичные значения L_0, T_0 выбираются при помощи уравнения линейной регрессии, а значение параметра S_0 обычно приравнивается к единице [13]. Эти модели хороши в прогнозировании значений и при этом их можно легко редактировать.

1.9 Метрики оценки точности прогноза

Подход к созданию модели и прогноза для других вариаций моделей в общем случае, одинаков и включает несколько этапов:

– поиск и скопление данных, и дальнейшее их изучение. Важно чтобы начальная информация была целостной, без недостающих элементов и опознавалась в общей единице измерений;

– подбор изначальных параметров модели. При этом рассматривается порядок авторегрессии, коэффициента сглаживания и скользящего среднего;

– на этом этапе необходимо разделить изначальные данные. Первая часть должна хранить внутри себя хотя бы 70-80% от общего количества имеющихся данных. Она предназначается для конфигурации модели, иными словами происходит поиск оценок коэффициентов. Другая половина изначальных данных – тестовое множество, которое применяется в проверке точности готовой модели. Завершенная модель способна максимально точно описывать исходные данные, но при этом она не будет считаться моделью для прогноза;

– следующим этапом является сравнение различных моделей по точности прогноза. Соответственно выбирается та у которой этот параметр является лучшим.

Для оценивания точности прогноза в одной модели одновременно используют разные метрики с индивидуальными свойствами.

В качестве примера возьмем идентифицированную модель временных рядов в которой уже построен прогноз. Получим вектор ошибок и представим его в виде разницы фактических и расчетных данных:

$$e = y - \hat{y}. \tag{34}$$

MFE – средняя ошибка прогноза. Демонстрирует приблизительное различие действительных(фактических) данных от прогнозируемых, при этом ошибки с противоположными значениями (положительными и отрицательными) взаимно сокращают друг друга. Прогноз будет более точным, если показатели выбранной метрики стремятся к нулю.

$$MFE = \frac{1}{n}\sum_{i=1}^{n} e_i. \quad (35)$$

MAE (MAD) – среднее абсолютное отклонение. Также, как и MFE отображает среднее абсолютное отклонение действительных данных от прогнозируемых. Единственным отличием от средней ошибки прогноза (MFE) ошибки с разными значениями не сокращают друг друга. Если значение метрики стремится к нулю, то прогноз будет более точным.

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |e_i|. \quad (36)$$

MAPE – средняя абсолютная ошибка в процентах. Демонстрирует процент отклонения действительных значений от прогнозируемых, но в этот вариант можно применять только для рядов, со средним значением больше единицы. Точность прогноза данной метрики зависит от её минимального значения.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n} \left|\frac{e_i}{y_i}\right| * 100\%. \quad (37)$$

MPE – средняя процентная ошибка. Метрика схожа с MAPE, но отличается тем, что положительные и отрицательные ошибки сокращают друг друга. Точность модели зависит от минимального процента ошибок.

$$MPE = \frac{1}{n}\sum_{i=1}^{n}\frac{e_i}{y_i} * 100\%. \quad (38)$$

MSE – средняя квадратическая ошибка. Демонстрирует среднее квадратическое отклонение действительных значений от прогнозируемых. Данная метрика зависит от её размера. Для более точного построения прогноза размер метрики должен быть минимален.

$$MSE = \frac{1}{n}\sum_{i=1}^{n} e_i^2. \quad (39)$$

Точность прогноза – это полностью противоположное определение, относительно ошибки прогноза. Его используют довольно редко. Это связанно с тем, что чаще всего для оценивания применяют совместно две ошибки MAPE и MAE. Оценка точности прогноза определяется при помощи разницы общего количества процентов данных и средней абсолютной ошибки в процентах.

При оценке достоверности прогнозирования созданной модели, как правило применяют метрики на выбор. Самыми часто используемыми метриками при сравнении разных моделей прогнозирования обычно берут среднее квадратическое отклонение и средняя абсолютная процентная ошибку.

Выводы по разделу. мы описали и изучили теоретическую информацию, которая необходима для выполнения программной реализации алгоритма выполнения прогнозирования временного ряда. Рассмотрены метрики оценки точности прогноза и выбран формат модели для реализации алгоритма.

2 Программная реализация

2.1 Необходимые программные средства

Чтобы произвести программную реализацию алгоритма прогнозирования временных рядов мы будем применять объектно-ориентированный язык программирования Python. Этот выбор осуществлен по следующим причинам:

– python как язык программирования довольно легко освоить;

– этот язык программирования является мульти платформенным;

– в Python содержится большая база библиотек, которая делает возможным работу с различными видами информации [18].

Также для удобства будем использовать облачный сервис Google Colab. Google Colab – это бесплатная интерактивная облачная среда для работы с кодом от Google.

Для того чтобы мы могли создавать модели прогнозирования и анализа временных рядов нам понадобятся следующие библиотеки:

– библиотека NumPy это библиотека для векторизованных вычислений. Она разработана на языке программирования C++ и это большой плюс. Нужна для применения массивов и матриц и включает в себя методы оптимизации численных вычислений и статические процедуры [4, 24];

– библиотека Matplotlib делает возможным применение функций для отображения многомерных графиков и диаграмм;

– высокоуровневая библиотека Pandas даёт возможность анализировать данные, построена она поверх низкоуровневой библиотеки NumPy. Помимо этого, позволяет считывать информацию из файлов типа: csv, excel, txt. Основными структурами данных в Pandas являются классы Series и DataFrame;

– библиотека Statsmodels предоставляет доступ к классам и функциям, благодаря которым появляется возможность оценивать разные статистические модели. Помимо этого, она позволяет проводить тесты с выводом статистики и их исследование. Результаты проверяются на соответствие существующим статистическим пакетам, чтобы убедиться в их правильности[24];

– библиотека Scipy при её активации, дает возможность работы с данными и рассматривать их со статистической точки зрения. «Благодаря данным библиотекам появляется возможность проводить различные тесты и строить графики корреляционных и автокорреляционных функций» [10].

Модель прогнозирования, которую мы реализуем будет представлена в модуле model.py.

В начале произведем загрузку данных для тестирования при помощи библиотеки Pandas. Это нужно для тестирования работы реализованной модели [2]. Будем пользоваться методом read_csv, так как изначально данные хранятся в формате csv. Сначала этому методу передают путь к файлу и знак разделения строк и формат цифр. После получения этой информации данные приводятся к формату ndarray при использовании метода to_numpy(), чтобы можно было работать с информацией в виде массивов.

Для графического отображения будем применять библиотеку matplotlib, а именно класс pyplot и метод plot(data). Этот метод берет заданный массив значений и применяет его для отображения на графике.

2.2 Реализация модели полиномиального тренда

Как мы уже решили в первом разделе, создаем модель полиномиального тренда. Эта модель будет представлена как класс PolynomialRegression. На рисунке 10 изображена UML–диаграмма этого класса.

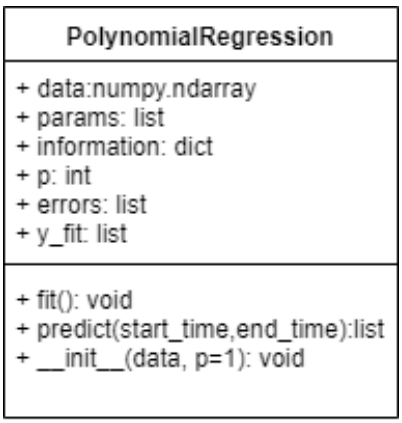

Рисунок 10 – Класс PolynomialRegression в UML-диаграмме

Данный класс содержит внутри себя следующие методы:

– _init_(data, p=1) – конструктор класса, работает путем того , что берет на вход одномерный массив и степень *p* полинома. Когда полином не имеет заданной степени, то в таком случае его значение приравнивается к 1;

– fit() – метод, предназначенный для поиска коэффициентов при помощи метода который находит оценки коэффициентов и производит подсчет метрик по методу наименьших квадратов;

– predict(start_time, end_time) – это метод, который берет два целых значения *start_time* и *end_time*, а потом строит прогноз в определенные моменты времени *t+start_time, t+start_time+1, …, t+end_time*.

Внутри класса PolynomialRegression хранятся следующие поля:

– data – это одномерный массив, внутри которого находится изначальный временной ряд;

– params – это коэффициенты полиномиального тренда, которые были вычислены при помощи метода наименьших квадратов;

– information – это словарь, внутри которого хранятся значения скорректированного коэффициента детерминации и нужных нам метрик *MAE, MSE, MAPE*;

– y_fit – это массив, который содержит все значения которые были найдены при помощи модели полиномиального тренда;

– errors – это массив, который хранит внутри себя остатки модели полиномиального тренда.

Перед тем как перейти к созданию модели, появляется необходимость в объекте класса PolynomialRegression. После этого необходимо произвести поиск коэффициентов регрессии с помощью метода fit(), а затем построить прогноз используя метод predict.

После этого мы можем реализовать проверку этого класса. Возьмем значения из диапазона [1, 2, 3, 4, 5] для параметра p. Далее производится отображение изначального временного ряда и новообретенного при помощи класса PolynomialRegression (рисунок 11).

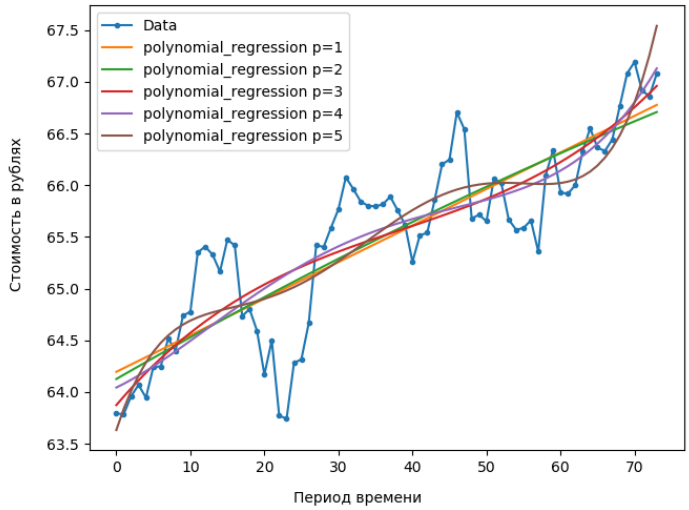

Рисунок 11 – Общий график изначального временного ряда и полиномиального тренда

Реализуем метод наименьших квадратов в методе least_square_method. Это нужно для того чтобы находить коэффициенты моделей. Он будет принимать на вход массивы *x* и *y*. Для нахождения обратной матрицы будем применять метод inv класса linalg из библиотеки numpy. В объекте класса ndarray хранит внутри себя атрибут T который содержит транспонированную матрицу. При помощи метода dot происходит умножение двух матриц.

Метрики для оценки точности модели были созданы в модуле my_statistics.py.

Провести анализ временого ряда и тестирование модели на остатки возможно при наличии библиотеки statsmodels, ведь там, как мы уже описали ранее, хранятся все требуемые для этого методы.

2.3 Реализация интерфейса

Когда у нас есть модель для прогнозирования и сглаживания, мы можем перейти к реализации интерфейса основной программы.

Программа будет содержать окно авторизации пользователей. Оно нужно для того чтобы конфиденциальная информация сотрудников не попала в чужие руки. Пример того как будет выглядеть окно авторизации представлен на рисунке 12.

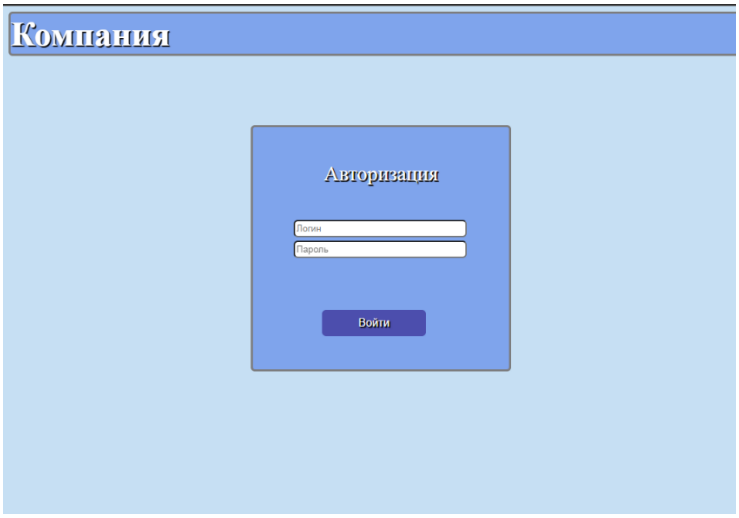

Рисунок 12 – Окно авторизации пользователей приложения

В программе будет две роли, а конкретно:

– администратор;

– сотрудник.

Администратору будет доступно свое меню, в котором будет содержаться информация о сотрудниках компании, а конкретно:

– инициалы сотрудника, заявившего о больничном;
– дата открытия больничного;
– дата закрытия больничного;
– отдел, в котором работает сотрудник компании;
– город, в котором работает сотрудник компании.

Помимо этого, панель администратора содержит панель, в которой можно сортировать информацию по дате открытия больничного с выбором интервала времени. Дни больничного можно указать самостоятельно или при помощи календаря. Также тут будет окно с графиком заболеваемости сотрудников и прогноза возможных рисков, который мы реализовали до этого. Пример интерфейса администратора изображен на рисунке 13.

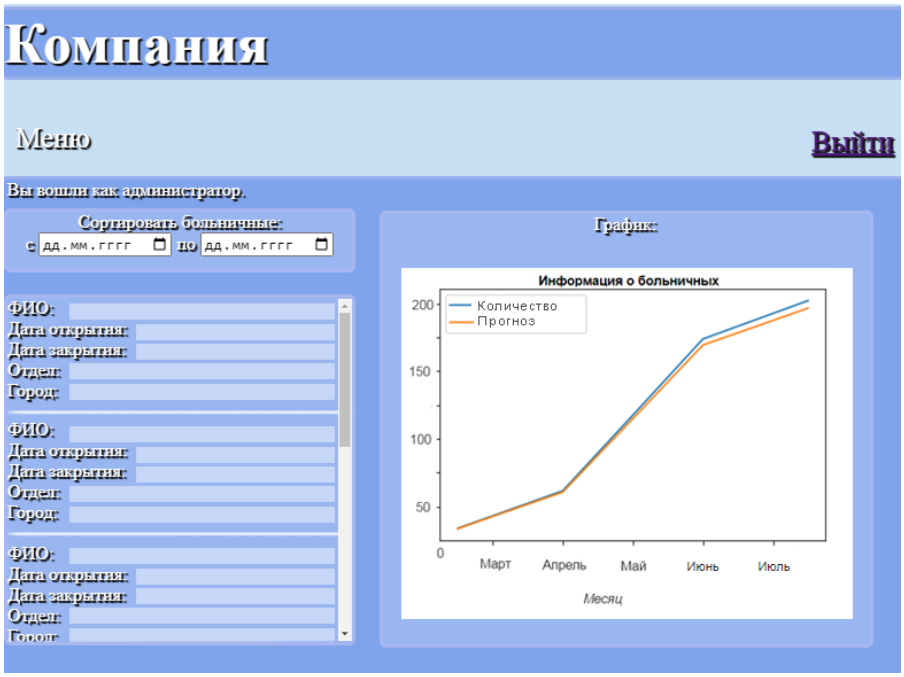

Рисунок 13 – Интерфейс администратора

Если авторизацию проходит сотрудник компании, интерфейс выглядит следующим образом. Так как приложение требуется для передачи

информации о своем заболевании, то интерфейс обычного пользователя не будет иметь много функций. В нем можно указать дату назначения своего больничного, а конкретно дату его начала и завершения. Дни больничного можно указать самостоятельно или при помощи календаря. После подтверждения данные передаются администратору и появляются в его интерфейсе. Пример того как выглядит интерфейс сотрудника представлен на рисунке 14.

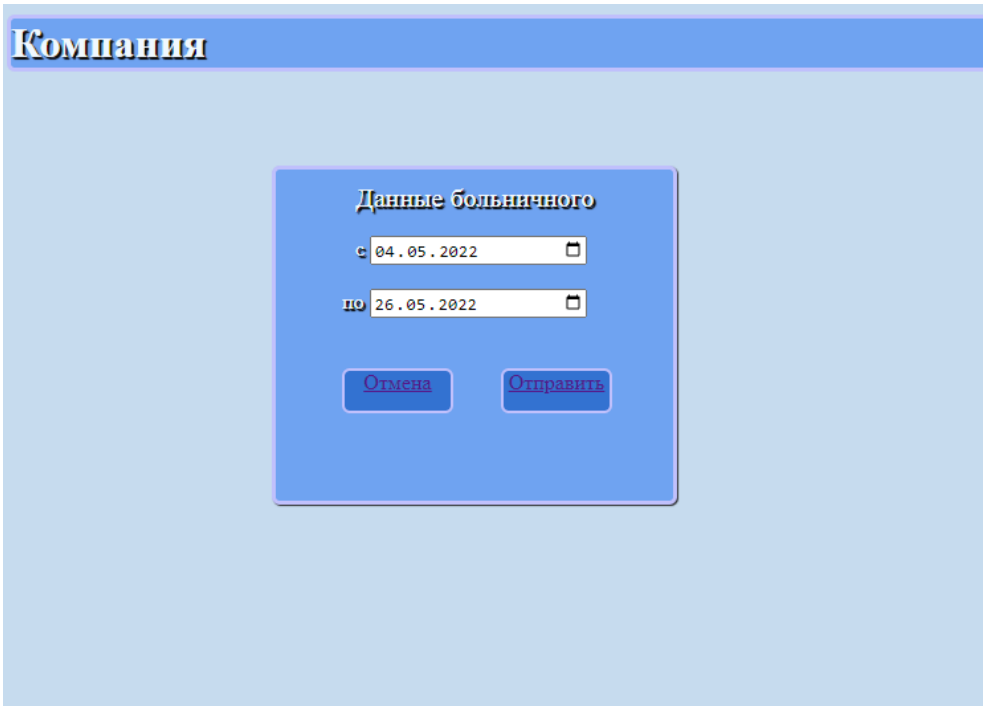

Рисунок 14 – Интерфейс сотрудника

Вывод по разделу: в данном разделе мы описали реализацию алгоритма для учета и анализа заболеваемости сотрудников компании при помощи прогнозирования временных рядов. Для реализации применили язык программирования Python. Использовались внутренние библиотеки pandas, matplotlib, NumPy, scipy и statsmodels. Также реализованы модели сглаживания и прогнозирования временных рядов. После полной реализации алгоритма и интерфейса можно приступить к проверке алгоритма и интерфейса.

3 Проверка алгоритма и интерфейса

3.1 Проверка алгоритма

Чтобы проверить работоспособность реализованного метода помимо нужных нам данных используем различные исходные данные в формате csv [25]. Брать их будем с сайта https://www.kaggle.com:

– прогнозирование заболеваемости сотрудников;

– прогнозирование прогулов сотрудников;

– цена на крипто валюту Doge Coin.

Первоначальной задачей для нас будет загрузка всех исходных данных и представление их в виде графиков временных рядов (рисунок 15).

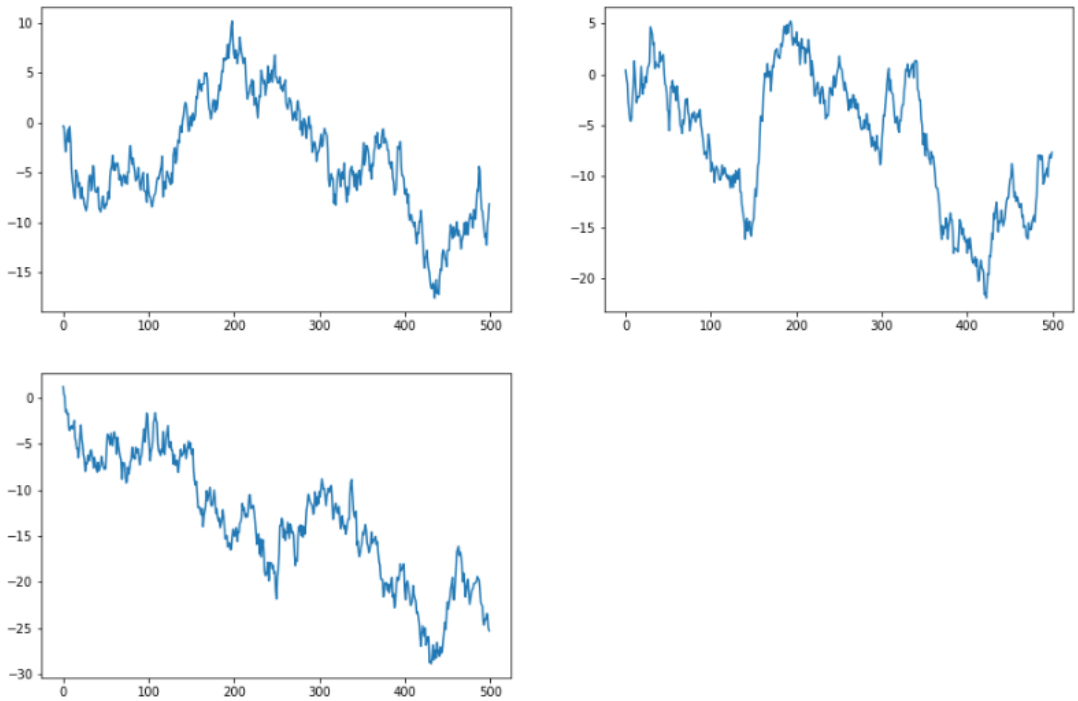

Рисунок 15 – Графики с исходными временными рядами

Проведя небольшой визуальный анализ всех трех графиков, можно сделать вывод о том, что тут нет ряда, обладающего стационарностью. Чтобы

это доказать, воспользуемся тестом Дики-Фуллера, который мы рассмотрели в первом разделе. Значения необходимой нам статистики *p* представлены ниже в таблице 3.

Таблица 3 – Значения *p* для исходных временных рядов

Временной ряд	Значение *p (p-value)*	Является ли ряд стационарным?
Прогнозирование заболеваемости сотрудников	0.1078657438022293	-
Прогнозирование прогулов сотрудников	0.8241697963183976	-
Цена на крипто валюту Doge Coin.	0.6018463069855985	-

Как мы уже узнали ранее, ряд не является стационарным если значение *p* больше 0,05. По итогу мы можем сказать, что среди изначальных временных рядов нет ни одного ряда которой являлся бы стационарным (рисунок 16).

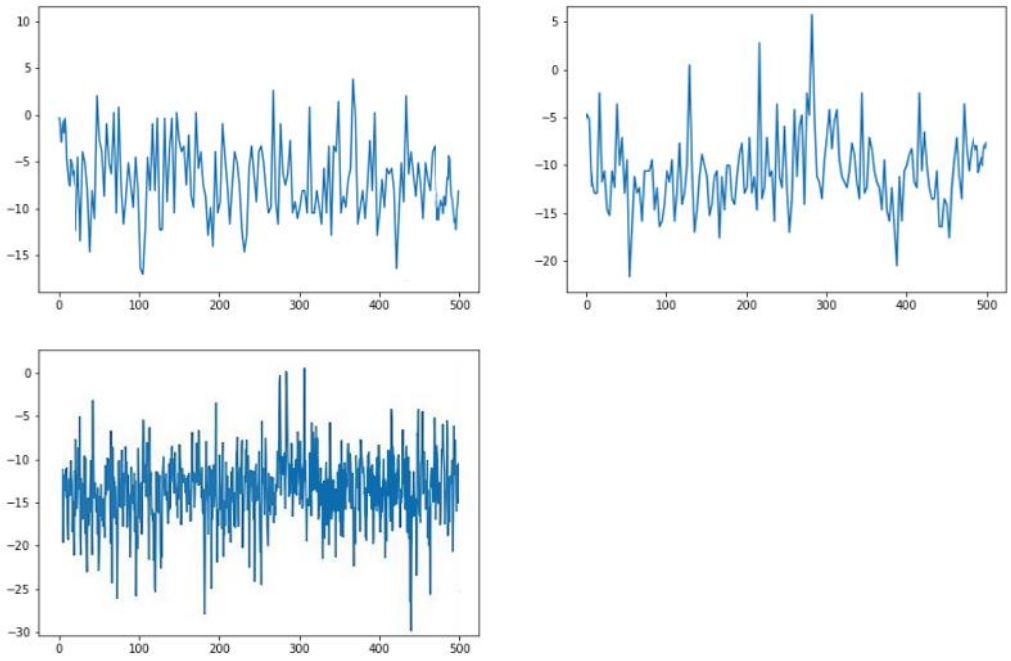

Рисунок 16 – Графики с интегрированными временными рядами

Таблица 4 – Значения *p* после интегрирования временных рядов

Временной ряд	Значение *p (p-value)*	Является ли ряд стационарным?
Прогнозирование заболеваемости сотрудников	2.13064805593792e-27	+
Прогнозирование прогулов сотрудников.	3.46518894587761e-10	+
Цена на крипто валюту Doge Coin.	3.98964510311376e-15	+

По итогу мы получили стационарность, путем того что один раз проинтегрировали изначальные временные ряды.

Далее нам потребуется разбить данные на две части. Первая часть будет обучающей выборкой, она будет содержать 80% от общего количества исходных данных. Вторая выборка для тестирования и соответственно будет содержать остальные 20%. Это нужно, чтобы построить и протестировать реализованную модель.

3.2 Выбор параметра для полиномиальной модели

Первым делом мы должны выбрать степень полиномиальной модели представив её в виде *p* и вычислить средний квадрат ошибки прогноза. Данные предоставлены в таблице 5.

Таблица 5 – Расчетные значения *MSE* для модели полиномиального тренда

Временной ряд	Степень модели (p)	MSE
Прогнозирование заболеваемости сотрудников	1	12.01
	2	1.93
	3	0.69
	4	4.02
	5	311.34
Прогнозирование прогулов сотрудников	1	0.165
	2	0.094
	3	1.84
	4	17.92
	5	6.8
Цена на крипто валюту Doge Coin.	1	15476.549
	2	74368.590
	3	18182.959
	4	4082.198
	5	1281496.956

Далее необходимо протестировать модель на адекватность. Это можно сделать в том случае, когда значение её коэффициента детерминации не меньше 0.5. Проверку можно провести при помощи значений скорректированного и стандартного коэффициента детерминации (таблица 6).

Таблица 6 – Показатели коэффициента детерминации

Временной ряд	Коэффициент детерминации	Скорректированный коэффициент детерминации
Прогнозирование заболеваемости сотрудников	0.5966	0.5885
Прогнозирование прогулов сотрудников	0.7521	0.7363
Цена на крипто валюту Doge Coin	0.7511	0.7461

Проверка дисперсии остатков модели наш следующий шаг. Делать это можно при помощи *F*-критерия Фишера и расчета *F*-значения, используя формулу:

$$FR = \frac{D\gamma}{Dad}, \tag{40}$$

где $D\gamma$ – дисперсия зависимой переменной.

$$D\gamma = \frac{\sum\left(\gamma - mean(\gamma)\right)^2}{N-1}, \tag{41}$$

где N – количество выбранных параметров;

Dad – дисперсия остатков модели.

$$D\nu = \frac{\sum(\gamma - \hat{y})^2}{N-k}, \tag{42}$$

где k – число параметров модели;

$\hat{y}$ –значения ряда, вычисленные внутри модели.

Далее происходит сравнение значения FR, которое мы получили при помощи формулы (40) с критическим значением критерия Фишера, с

доверительной вероятностью 95%, степенями свободы – *N-1, N-k*. Это происходит при помощи функции criterion_fisher. Она возвращает значение FR которое мы нашли до этого и значение критерия Фишера, взятое из таблицы. Результаты расчетов приведены в таблице 7.

Таблица 7 – Значения F-критерия

Временной ряд	Расчетный F-критерий	Максимальное значение
Прогнозирование заболеваемости сотрудников	2.455	1.261
Прогнозирование прогулов сотрудников	3.953	1.604
Цена на крипто валюту Doge Coin	3.956	1.2669

Судя по данным, полученные значения критерия F оказались больше минимальных табличных значений. Это обозначает, что с вероятностью 95% созданные модели можно называть статистически адекватными.

Последним шагом станет проверка остатков на отсутствие автокорреляции при помощи критерия Дарбина-Уотсона. Рассчитаем d-значение, используя формулу:

$$d = \frac{\sum_{i=1}^{N-1}(e_i - e_{i+1})^2}{\sum_{i=1}^{N} e_i^2}, \quad (43)$$

где e_i – остатки модели.

Данные критериевd_L и d_U мы возьмем из таблицы с числом наблюдений N, количеством параметров модели k и уровнем значимости α.

Если d меньше нижней границы d_L, то гипотеза о независимости случайных величин отклоняется и в остатках присутствует корреляция.

Если d больше верхней границы, d_U то гипотеза не отклоняется.

Если d находится между d_L и d_U, то тест некорректен.

Когда d превышает показатель 2, то сравнивается значение $(4 - d)$.

В том случае если все новообретенные показатели стали меньше нижней границы тех значений, которые были изначально, тогда мы можем не брать во внимание гипотезу о независимости остатков с вероятностью 95%.

Рассчитанные значения d для всех трех моделей представлены на таблице 8.

Таблица 8 – Значения d-критерия в остатках модели.

Временной ряд	Расчетный d-критерий	d_L	d_U
Прогнозирование заболеваемости сотрудников	0.1314	1.73	1.79
Прогнозирование прогулов сотрудников	0.411	1.462	1.624
Цена на крипто валюту Doge Coin	0.18	1.728	1.810

Смотря на показатели из таблицы можно сказать, что в остатках все еще находится ценная информация о временных рядах. Соответственно можно заявить о готовности нашего алгоритма.

3.3 Осмотр интерфейса

Далее рассмотрим возможности интерфейса. В панели сотрудника не так много функций, но проверить работоспособность данного модуля необходимо. Попробуем подать заявку на больничный. Процесс представлен на рисунках 17-18.

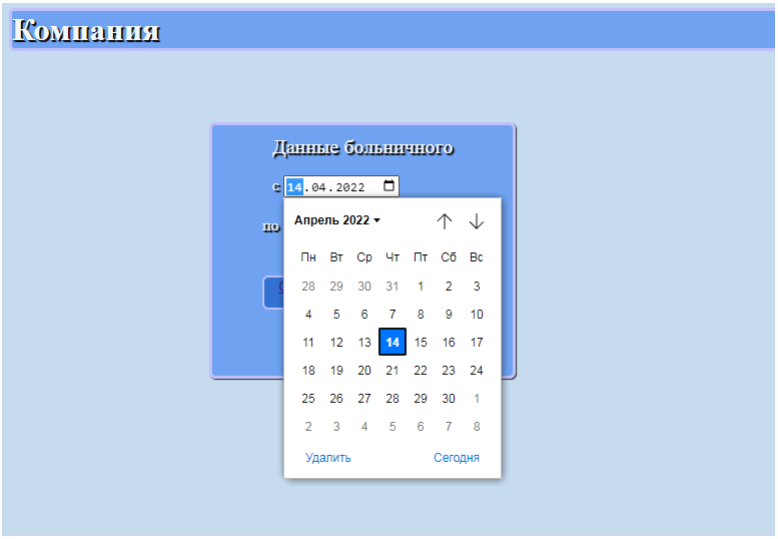

Рисунок 17 – Выбор даты для больничного

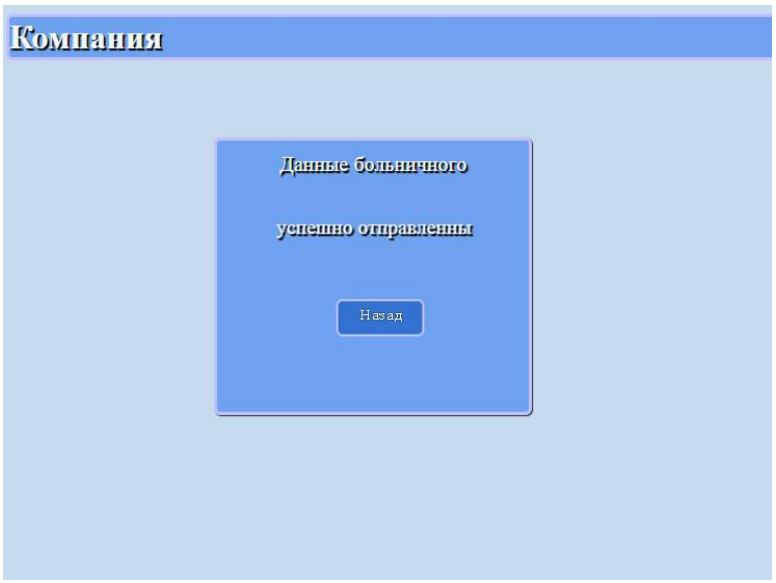

Рисунок 18 – Заявка подана

Проверим сортировку по датам среди сотрудников, подавших заявления на больничный в марте в режиме администратора (рисунок 19).

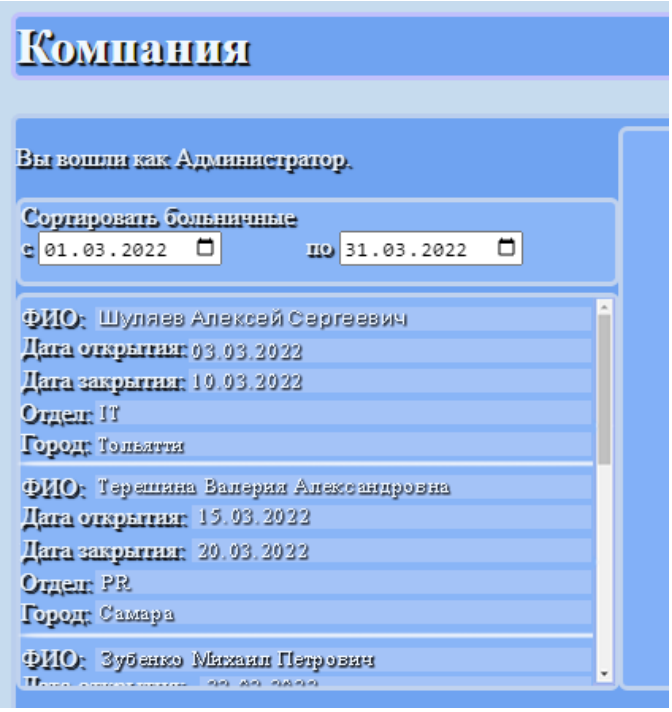

Рисунок 19 – Список больничных за март

Вывод по разделу. Подводя итог по данному разделу, можно сказать, что были выполнены следующие задачи:

– проверен разработанный алгоритм прогнозирования при помощи различных входных данных;

– подобран необходимый параметр для разработанной модели;

– проверен разработанный интерфейс и продемонстрированы его способности.

В результате наблюдений за разработанным алгоритмом, можно заявить о его успешной работоспособности.

Заключение

Во время написания бакалаврской работы мы рассмотрели множество теоретических материалов о анализе и прогнозировании временных рядов, а еще различные методы оценки адекватности построенной модели и метрики оценки точности прогноза.

В ходе работы был выполнен ряд следующих задач:

– проведено исследование предметной области;

– рассмотрены различные модели прогнозирования;

– рассмотрены различные метрики для построения моделей прогнозирования;

– найдена подходящая модель прогнозирования для поставленного случая;

– реализован алгоритм для прогнозирования заболевания;

– реализован интерфейс приложения и выбранный метод на языке программирования Python с использованием облачного сервиса Google Colab;

– проведена проверка интерфейса и работоспособности разработанного метода при помощи различных метрик.

В результате выполнения бакалаврской работы были исследованы способы прогнозирования распространения вирусов и разработано приложение для учета и анализа динамики заболеваний у сотрудников компании. При помощи использования данного приложения, руководство компании получит возможность делать прогнозы вспышек заболеваемости у сотрудников компании и на основе полученных данных, они будут принимать дальнейшие решения о переходе на дистанционных режим работы.

В выпускной квалификационной работе раскрывается значимость и актуальность исходного положения: разработки алгоритма прогнозирования и анализа заболеваемости сотрудников компании.

В результате написания работы была полностью раскрыта тема работы, достигнута цель бакалаврской работы

Список используемой литературы

Научная и методическая литература

1. Арефьева Н.Т. Прогнозирование и его социокультурные цели / Н.Т. Арефьева // Электронный журнал «Знание. Понимание. Умение». – 2010. – № 4 – С. 1.
2. Бринк *Х.,* Ричардс Д., Феверолф М. Машинное обучение. — Питер, 2018. — 336 с. — ISBN 978-5-496-02989-6.
3. Верещагин Н.К., Щепин Е.В. Информация, кодирование и предсказание. – М.: ФМОП, МЦНМО, 2012. – 238 с.
4. Воронцов К. В., «Комбинаторный подход к оценке качества обучаемых алгоритмов», Математические вопросы кибернетики, 13, ред. О. Б. Лупанов, Физматлит, М., 2004, 5-36 mathscinet;
5. Дуброва Т.А. — Москва: ЮНИТИ-ДАНА, 2003. — ISBN 5-238-00497-4.
6. Каток А.Б., Хассельблат Б. Введение в современную теорию динамических систем / Пер. с англ. М.: Факториал УРСС, 1999. 767 с.
7. Коломыйцев О.А. Обзор методов и подходов к прогнозированию финансовых временных рядов / О.А. *Коломыйцев, В.Ю. Шелепов //* Сборник научных трудов по материалам международной научно-практической конференции. – 2008. – Т. 8. № 1 – С. 71-73.
8. Лоскутов А.Ю., Михайло А.С. Основы теории сложных систем. – М.-Ижевск: Институт компьютерных исследований, 2007. – 620 с.
9. Мишулина О.А. Статистический анализ и обработка временных рядов. – М.: МИФИ, 2004. – С. 180. – ISBN 5-7262-0536-7.
10. Нуньес-Иглесиас Х., Уолт Ш., Дэшноу Х. Элегантный SciPy = Elegant SciPy. — ДМК Пресс, 2018. — 266 с. — ISBN 978-5-97060-6001.
11. Орлова И.В. Экономико-математические методы и модели: компьютерное моделирование: учебное пособие / И.В. Орлова, В.А. Половников. – Москва: Вузовский учебник, 2007. – 365 с.

12. Рашка С. Python и машинное обучение / С. Рашка; пер. с англ. А.В. Логунова. – Москва: ДМК Пресс, 2017. – 418 с.
13. Сажин Ю.В. Анализ временных рядов и прогнозирование: учебник / Ю. В. Сажин, А. В. Катынь, Ю. В. Сарайкин. – Саранск: Изд-во Мордов. ун-та, 2013. – 192 с.
14. Светуньков И.С. Методы социально-экономического прогнозирования. Том 1. Теория и методология. / И.С. Светунькова. – Москва: Юрайт, 2015. – 351 с.
15. Трегуб А.В. Методика построения модели ARIMA для прогнозирования динамики временных рядов / А. В. Трегуб // Лесной вестник. – 2011. – № 5. – С. 179-183.
16. Уткин В.Б. Эконометрика / В.Б. Уткин. – Москва: Дашков и К, 2017. – 564 с.
17. Федосеев, В.В, А.Н. Гармаш, И.В. Орлова. Экономико-математические методы и прикладные модели: учебник для бакалавров: – М. Юрайт, 2012. – 328 с.
18. Федоров, Д. Ю. Программирование на языке высокого уровня Python. — Москва: Издательство Юрайт, 2022. — 210 с. — ISBN 978-5-534-14638-7.
19. Чураков Е.П. Прогнозирование эконометрических временных рядов: учебное пособие / Е.П. Чураков. – Москва: Финансы и статистика, 2008. – 208 с.
20. Dettlng M. Applied Time Series analysis / Dr. M. Dettling: ETH. – 2014. – 176 p.
21. Faraway J. Linear Models with R. / J.J. Faraway: Chapman & Halll/CRC. – 2009. – 255 p.
22. Robert H. Time series analysis and its applications, 3rd Edition / R.H. Shumway, D.S. Stoffer: Springer New York. – 2011. – 202 p.
23. McKinney W. Python for Data Analysis, second Edition / W. McKinney: O'Reilly Media. – 2018. – 541 p.

Электронные ресурсы

24. Hyndman R.J Forecasting: principles and practice, 2nd Edition [Электронный ресурс] / Melbourne, Australia. – Режим доступа: https://otexts.com/fpp2/
25. Kaggle’s Documentation [Электронный ресурс]. – Режим доступа: https://www.kaggle.com/.