МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение

высшего образования

«ТЮМЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ШКОЛА КОМПЬЮТЕРНЫХ НАУК

Кафедра программного обеспечения

**ОТЧЁТ ПО ПРОЕКТНОЙ ДЕЯТЕЛЬНОСТИ**

Тема: «Расширение инструментальной системы StatMedLab для статистической обработки результатов биохимических измерений в медицине»

|  |  |  |
| --- | --- | --- |
| Выполнили работу  студенты 4 курса |  | Ануфриев Никита Александрович    Василец Анастасия Артёмовна |
| Руководитель  Доцент, к.ф.-м.н. |  | Ступников Андрей Анатольевич |

Тюмень - 2024

# СОДЕРЖАНИЕ

[**СОДЕРЖАНИЕ 2**](#_Toc185194033)

[**ВВЕДЕНИЕ 3**](#_Toc185194034)

[**Глава 1. Описание приложения 5**](#_Toc185194035)

[1.1. Текущая версия приложения 5](#_Toc185194036)

[1.2. Идея решения 6](#_Toc185194037)

[1.3. Требования 6](#_Toc185194038)

[1.4. Ограничения 7](#_Toc185194039)

[**Глава 2. Используемые методы и технологии 8**](#_Toc185194040)

[2.1. Временные ряды 8](#_Toc185194041)

[2.2. Технологии 12](#_Toc185194042)

[2.2.1. Технологии текущей версии приложения 12](#_Toc185194043)

[2.2.2. Технологии текущей версии приложения 12](#_Toc185194044)

[**Глава 3. Реализация 13**](#_Toc185194045)

[3.1. Загрузка новых данных 13](#_Toc185194046)

[3.2. Определение пола пациента по ФИО 17](#_Toc185194047)

[3.3. Анализ временных рядов 20](#_Toc185194048)

[3.4. Сохранение параметров выборок и исследований 22](#_Toc185194049)

[3.5. Тестирование 23](#_Toc185194050)

[3.6. Добавленный функционал 23](#_Toc185194051)

[**ЗАКЛЮЧЕНИЕ 25**](#_Toc185194052)

[**СПИСОК ЛИТЕРАТУРЫ 26**](#_Toc185194053)

# ВВЕДЕНИЕ

Целью статистического анализа биохимических данных крови является извлечение значимой информации, которая может существенно улучшить процессы диагностики, мониторинга лечения и прогноза заболевания [2, 4].

Основные задачи, стоящие перед исследователями, включают в себя несколько этапов:

1. Формулирование гипотез и вопросов исследования.
2. Планирование и сбор данных, в том числе определение ключевых биохимических показателей, необходимых для анализа, установление размера выборки и контроль качества данных на этапе сбора.
3. Обработка и очистка данных. Этот этап включает в себя выявление и исправление пропущенных значений, а также обработку выбросов, которые могут исказить результаты анализа.
4. Выбор и применение соответствующих статистических методов.
5. Интерпретация результатов анализа. Необходимо оценить как статистическую, так и клиническую значимость полученных результатов.
6. Визуализация результатов, их представление в виде таблиц и графиков.

Эффективный статистический анализ биохимических данных крови требует глубоких знаний статистических методов и биохимических процессов, а также навыков работы со специализированными программными пакетами для обработки и анализа данных.

Основной проблемой, которое должно было решить приложение, является то, что в данные по медицинским исследованиям в Тюменской области хранятся с разной структурой в разных районах. У медицинских исследователей нет возможности собирать эти данные воедино и проводить их статистическую обработку. Также, большинство программ, реализующих методы статистической обработки (SPSS, Stata. Rlang), требуют от пользователя знания специальных команд или одного из языков программирования, у них нет интеграции с МКБ-10.

Исходя из обозначенных проблем, была актуальна разработка программного обеспечения, способного агрегировать и обрабатывать результаты биохимических анализов с разных медицинских учреждений.

Приложение было разработано и развёрнуто в рамках прошлой ВКР, сейчас оно работает в открытом доступе и им пользуются врачи-эпидемиологи. Было реализовано большое количество функций статистической обработки данных, но малое количество данных сократило этот список. Для прошлой ВКР заказчиком были предоставлены результаты анализов за полгода, что не позволяло выявлять временную динамику.

Остались некоторые нерешённые проблемы с пользовательским опытом приложения. В текущей версии не сохраняются параметры выборок и исследований, созданных пользователем, то есть, при каждой перезагрузке страницы или повторном входе в систему все действия удаляются. Также пользователи не могут сохранить результаты проведённого исследования в локальный файл, который можно изменять. Так как приложение используется при написании научных статей, у пользователей должна быть возможность изменять шрифт в таблицах и графиках, которые являются результатами исследований.

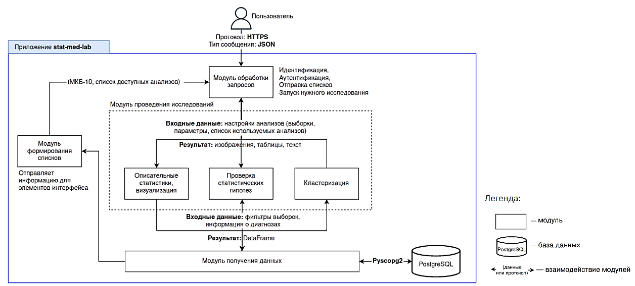
Выделенные в текущей версии приложения проблемы ограничивают аналитические возможности системы и не позволяют пользователю беспрепятственно использовать его возможности.

# Глава 1. Описание приложения

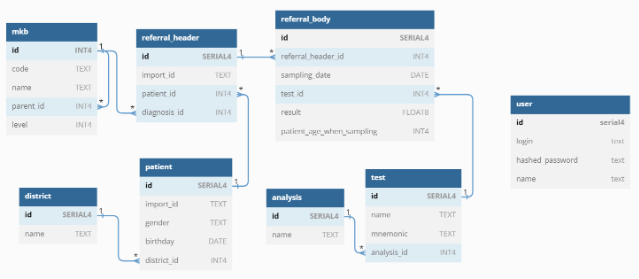
## 1.1. Текущая версия приложения

Текущая версия приложения состоит из 4 основных модулей (рис.1):

1. База данных PostgreSQL - в ней хранятся данные о результатах анализов пациентов. Также в ней хранится информация о пользователях приложения (рис.2).
2. Модуль импорта данных – в нём происходит обработка и загрузка данных в базу данных.
3. Серверная часть приложения - в ней поступают запросы на проведение методов статистического анализа по определённой выборке.
4. Клиентская части приложения - пользовательский интерфейс в виде веб-сайта.



*Рис. 1. Архитектура приложения*

.

*Рис. 2. ER-диаграмма базы данных*

Сейчас приложение позволяет указывать пол, возраст и диагноз пациентов, район и диапазон дат получения результата анализа при создании выборки для исследования. Самих исследований есть множество видов [3]:

1. Базовые описательные статистики.
2. Графики изучения распределения – гистограмма распределения, график плотности, ящичная диаграмма, скрипичная диаграмма.
3. Изучение корреляции - корреляционный анализ, сетка шестиугольников, диаграмма рассеяния.
4. Оценка статистических гипотез – оценка значимости коэффициента корреляции, сравнение генерального среднего с константой, сравнение средних значений двух независимых выборок, медианный критерий, однофакторный дисперсионный анализ.
5. Кластеризация методом k-средних.

## 1.2. Идея решения

Для решения указанных проблем можно сделать следующее:

1. Добавить в приложение методы анализа временных рядов, они позволяют выделить тренд и сезонные закономерности в изменениях результатов анализов. Так как для реализации этого проекта заказчиком был предоставлен датасет с более широким промежутком времени (с 2018 по середину 2024) года, данных должно быть достаточно для корректной работы временных рядов.
2. Сохранение параметров выборок и исследований предполагается хранить в базе данных, чтобы пользователь мог загрузить в текущую сессию только нужные ему выборки и исследования.
3. Изменение внешнего вида результатов исследований можно сделать вне основного приложения. Предполагается, что пользователь сможет сохранять результаты исследований в локальный файл изменяемого формата (например, .docx).

## 1.3. Требования

Общие требования к приложению были описаны заказчиком при начале работы над новой версией:

1. Приложение должно уметь обрабатывать большие объёмы данных и возвращать результаты статистических анализов по ним.
2. Приложение должно иметь графический интерфейс.
3. Приложение должно иметь интеграцию с международным классификатором заболеваний МКБ-10.
4. Должна сохраняться конфиденциальность личных данных пациентов, переданная заказчиком информация должна быть обезличена.

Проектной командой были определены требования к тому функционалу, который должен быть реализован в рамках проекта:

1. Приложение должно содержать методы статистической обработки временных рядов, что включает в себя: сглаживание графика временного ряда, выделение тренда, выделение сезонных закономерностей.
2. Приложение должно сохранять информацию о параметрах выборок и исследований пользователя в базу данных.
3. Приложение должно давать возможность пользователю сохранять результаты исследования на локальное устройство в виде отчёта в изменяемом формате.

## 1.4. Ограничения

Основными ограничениями проекта являются используемые данные и время на разработку:

1. Разрешено использовать только те данные, которые уже были загружены в базу данных приложения в рамках прошлой ВКР, и новые данные, предоставленные заказчиком.
2. Время на разработку ограничено одним семестром, с середины сентября по конец декабря 2024 года.

# Глава 2. Используемые методы и технологии

## 2.1. Временные ряды

Анализ временных рядов – это статистическая методология, используемая для изучения данных, собранных в течение определенного периода времени. Цель анализа – выявление закономерностей, трендов, сезонности и других особенностей во временных данных, а также построение прогнозов на будущее. Методы анализа временных рядов можно классифицировать по нескольким критериям, например, по типу моделируемых процессов (стационарные/нестационарные), по подходу к анализу (детерминистский/стохастический) и по используемым инструментам (классическая статистика, машинное обучение).

В контексте медицинской статистики, анализ временных рядов часто используется для мониторинга динамики показателей здоровья пациентов (например, уровня глюкозы в крови, артериального давления), оценки эффективности лечения, исследования сезонности заболеваемости и т.д. [5]

Ниже представлены основные категории методов, используемых в анализе временных рядов, с акцентом на их применение в медицинской статистике:

* + - 1. Методы сглаживания: Цель – уменьшить влияние шума (случайных флуктуаций) и выявить основную тенденцию.
         1. Скользящие средние (Moving Average): этот метод вычисляет среднее значение за определённый период (окно) наблюдений. Окно "скользит" по ряду, и для каждого момента времени вычисляется среднее значение по соответствующему окну.

Преимущества: Простота реализации и интерпретации.

Недостатки: Задержка реакции на резкие изменения тренда (чем шире окно, тем больше задержка). Потеря крайних значений ряда, соответствующих размеру окна. Различные типы окон (прямоугольное, треугольное, Гаусса) позволяют регулировать чувствительность к выбросам.

* 1. Экспоненциальное сглаживание (Exponential Smoothing): в отличие от скользящего среднего, экспоненциальное сглаживание придает больший вес недавним наблюдениям. Это позволяет быстрее реагировать на изменения тренда. Существуют различные модификации:
     1. Простое экспоненциальное сглаживание. Подходит для рядов без тренда и сезонности.
     2. Двойное экспоненциальное сглаживание (Holt). Учитывает линейный тренд.
     3. Тройное экспоненциальное сглаживание (Holt-Winters). Учитывает как линейный тренд, так и сезонность. Может быть аддитивным (сезонность добавляется к тренду) или мультипликативным (сезонность умножается на тренд).

Преимущества: Более плавное сглаживание, чем скользящее среднее, лучшее реагирование на изменения тренда.

Недостатки: Требует подбора параметров сглаживания (например, коэффициента сглаживания α), что может быть сложной задачей.

1. Декомпозиция временного ряда: Разложение ряда на составляющие компоненты: тренд (долгосрочная тенденция), сезонность (периодические колебания) и остаток (случайная составляющая). Это позволяет анализировать каждую составляющую отдельно.
   1. Аддитивная модель: Y = T + S + R, где Y – исходное значение, T – тренд, S – сезонность, R – остаток. Подходит, если амплитуда сезонных колебаний не зависит от уровня тренда.
   2. Мультипликативная модель: Y = T \* S \* R. Подходит, если амплитуда сезонных колебаний пропорциональна уровню тренда.

Преимущества: Позволяет получить более глубокое понимание структуры временного ряда.

Недостатки: Может быть сложно определить, какая модель (аддитивная или мультипликативная) лучше подходит для конкретных данных.

1. Автокорреляционный и частичный автокорреляционный анализ: Используется для выявления автокорреляции (корреляции между значениями ряда, отстоящими друг от друга на определенное число периодов – лаг) и для определения порядка авторегрессионных моделей.
   1. Автокорреляционная функция (ACF): Показывает корреляцию между значениями ряда и их сдвинутыми копиями.
   2. Частичная автокорреляционная функция (PACF): Показывает корреляцию между значениями ряда, с учётом влияния промежуточных значений.
2. Вычисления тенденций и сезонных закономерностей в данных охватывает несколько ключевых аспектов, которые помогают анализировать временные ряды и делать прогнозы на основе исторических данных.
   1. В первую очередь, важно определить, что такое тенденция. Тенденция представляет собой общее направление, в котором движется ряд данных за определённый период времени. Тенденции могут быть различными: восходящими, когда значения данных увеличиваются со временем; нисходящими, когда значения данных уменьшаются со временем, что может свидетельствовать о снижении интереса, ухудшении экономической; и стационарными, когда значения данных колеблются вокруг постоянного уровня, что может указывать на стабильность в определённой области без значительных изменений.
   2. Следующим важным аспектом являются сезонные закономерности. Сезонные закономерности представляют собой регулярные колебания в данных, которые происходят в определённые временные интервалы, такие как ежемесячно, ежеквартально или ежегодно. Эти закономерности могут быть вызваны различными факторами.

Для анализа тенденций и сезонных закономерностей. Используются графики и диаграммы позволяют наглядно увидеть, как данные изменяются со временем, и помогают выявить как долгосрочные тенденции, так и краткосрочные колебания. Визуализация может включать линейные графики, столбчатые диаграммы и другие формы представления данных, которые делают информацию более доступной для анализа.

* 1. Другим важным методом является использование скользящих средних. Скользящие средние помогают сгладить временные ряды, убирая краткосрочные колебания и позволяя лучше выявить долгосрочные тенденции. Этот метод работает путём вычисления среднего значения данных за определённый период, что позволяет уменьшить влияние случайных колебаний и выделить более устойчивые тренды.
  2. Метод наименьших квадратов (МНК) представляет собой один из наиболее широко используемых статистических методов для анализа временных рядов, позволяя выявлять как тенденции, так и сезонные закономерности в данных. Этот метод основывается на принципе минимизации суммы квадратов отклонений между наблюдаемыми значениями и значениями, предсказанными моделью. Это позволяет находить наилучшие параметры для модели, которая описывает зависимость между переменными.

Основная идея метода заключается в том, чтобы минимизировать сумму квадратов ошибок, то есть разностей между фактическими значениями и предсказанными значениями. В контексте временных рядов, где данные зависят от времени, МНК может быть использован для построения линейной регрессионной модели. Эта модель описывает, как зависимая переменная, такая как продажи, температура или уровень безработицы, изменяется во времени.

Для начала анализа необходимо определить модель, которая будет использоваться для описания данных. В простейшем случае это может быть линейная модель, которая имеет вид:

где — это значение зависимой переменной в момент времени , — это свободный член, который представляет собой значение зависимой переменной, когда независимая переменная равна нулю, — это коэффициент наклона, который показывает, как изменяется при изменении , а — это ошибка предсказания, отражающая влияние всех факторов, не учтённых в модели.

Для нахождения коэффициентов и необходимо минимизировать функцию потерь, которая представляет собой сумму квадратов ошибок:

где — это общее количество наблюдений. Для нахождения минимальных значений и необходимо взять производные функции потерь по каждому из коэффициентов, приравнять их к нулю и решить систему уравнений. Это приведёт к получению оценок для коэффициентов, которые минимизируют сумму квадратов ошибок. В результате, полученные значения и будут представлять собой наилучшие оценки для линейной зависимости между переменными.

Однако временные ряды часто содержат не только тренды, но и сезонные компоненты. Сезонные закономерности представляют собой регулярные колебания в данных, которые происходят в определённые временные интервалы, такие как месяцы, кварталы или годы. Эти закономерности могут быть вызваны различными факторами, включая сезонные изменения в погоде, праздники и экономические циклы.

Для учёта сезонных закономерностей в модели можно использовать расширенные подходы, такие как сезонная декомпозиция или модели с сезонными эффектами. Одним из распространённых методов является добавление сезонных индикаторов в модель. Например, если данные имеют годовой сезонный цикл, можно добавить переменные, которые принимают значения, соответствующие каждому сезону. Это может выглядеть следующим образом:

где — это индикаторы для каждого сезона. Эти индикаторы позволяют модели учитывать влияние сезонных факторов на зависимую переменную.

## 2.2. Технологии

### 2.2.1. Технологии текущей версии приложения

Для разных модулей приложения использовались следующие технологии:

1. Модуль базы данных. В качестве СУБД была использована PostgreSQL.
2. Модуль импорта данных:
   1. Язык программирования Python.
   2. Библиотеки numpy и pandas для получения данных из Excel-файлов и их предобработки.
   3. Библиотека psycopg2 для загрузки данных в базу данных.
3. Серверная часть приложения:
   1. Язык программирования Python.
   2. Библиотека aiohttp для разработки API и серверной логики приложения.
   3. Библиотеки scikit-learn и scipy для проведения статистических исследований на полученных данных.
   4. Библиотеки matplotlib и seaborn для визуализации результатов исследований.
   5. Формат данных JSON для передачи данных между серверной и клиентской частью приложения.
4. Клиентская часть приложения. Пользовательский интерфейс был разработан с помощью фреймворка React.

### 2.2.2. Технологии текущей версии приложения

Для сохранения целостности приложения было решено сократить количество новых использованных технологий до минимума, но для сохранения параметров выборок и исследований было решено использовать документоориентированную СУБД MongoDB, причина добавления нового вида базы данных будет описана позже.

# Глава 3. Реализация

## 3.1. Загрузка новых данных

Как упоминалось ранее, для разработки новой версии приложения команде были выданы новые данные по биохимическим измерениям крови в период с 2018 по середину 2024 года по городу Тюмень.

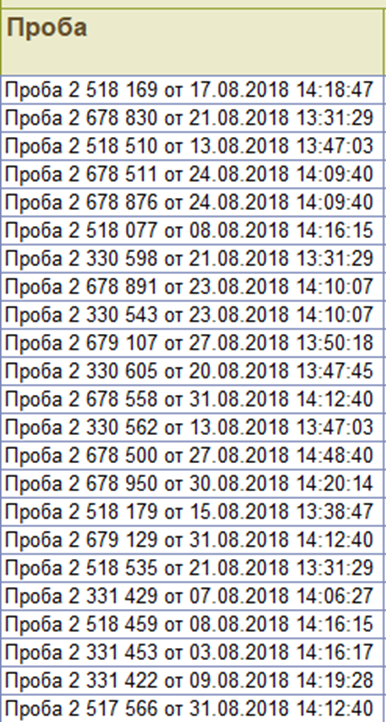
Структура базы данных существенно не изменялась с предыдущей версии приложения, в некоторые таблицы было добавлено ограничение уникальности на некоторые столбцы, чтобы можно было использовать оператор ON CONFLICT для СУБД PostgreSQL.

Новые данные представляют собой Excel-таблицы с такой структурой по столбцам:

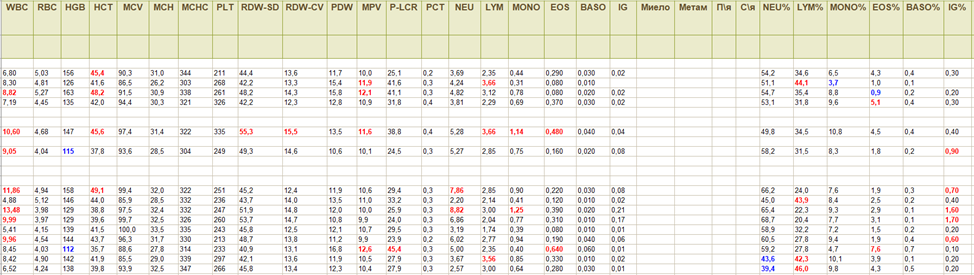
1. Название пробы – в ней хранится номер пробы и дата проведения измерений (рис. 3);
2. ФИО пациента;
3. Дата рождения пациента;
4. Поставленный пациенту диагноз, который был поставлен на основе результатов измерения.

В остальных столбцах прописаны результаты измерений по разным показателям, в них присутствуют результаты и для общего, и для биохимического анализа крови (рис. 4).

Также датасет был разделён на папки по годам и на месяцы по Excel-файлам.



*Рис. 3. Столбец с описанием пробы*



*Рис. 4. Результаты биохимических измерений*

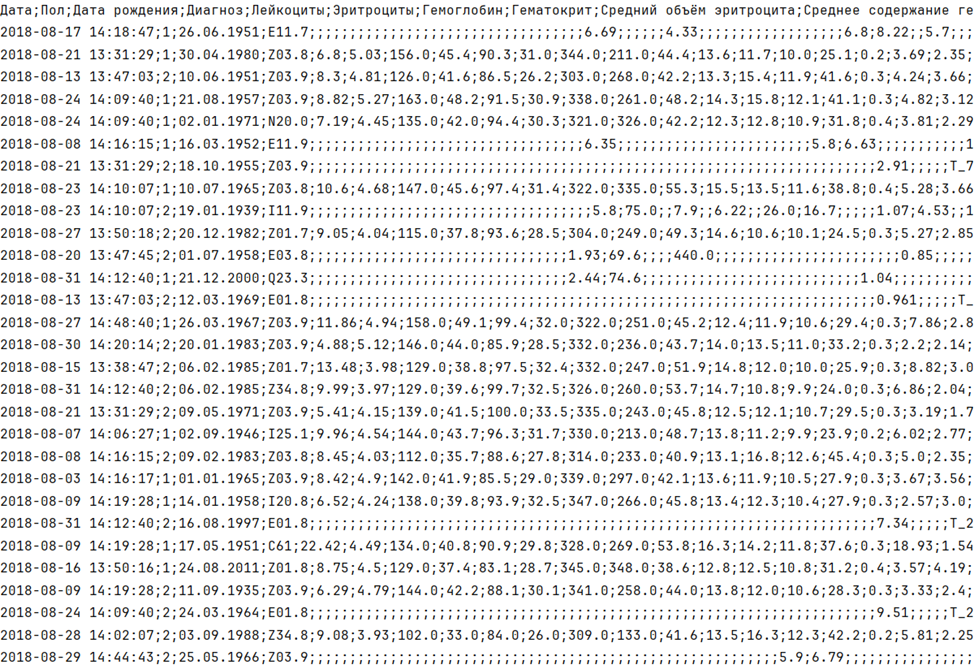
Так как в датасете было более 100 параметров результатов измерений, было решено отсеять некоторые из них. Те параметры, количество ненулевых записей в которых менее 2% от всех записей во всех файлах, были исключены.

Полученные данные были конфиденциальными, поэтому их нужно было деперсонализировать, для этого нужно было определить пол пациента по его ФИО, это важный параметр в исследованиях.

Полная предобработка включала в себя следующие действия:

1. Выделение даты из столбца «Проба» (рис. 3).
2. Очистка строк изначальной таблицы, если в основных столбцах находятся пустые значения.
3. Присвоение пациентам ID по их ФИО и дате рождения.
4. Построение ID измерения из префикса, тела (номера в таблице) и постфикса.
5. Определение пола пациента по ФИО с помощью предобученной модели.

Предобработанные данные загружаются в CSV-файлы (рис. 5).

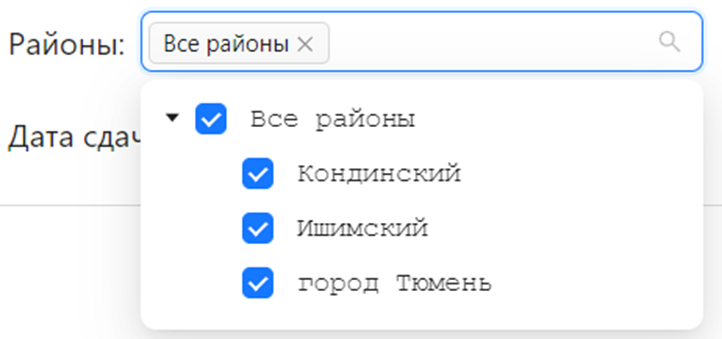


*Рис. 5. Предобработанный CSV-файл*

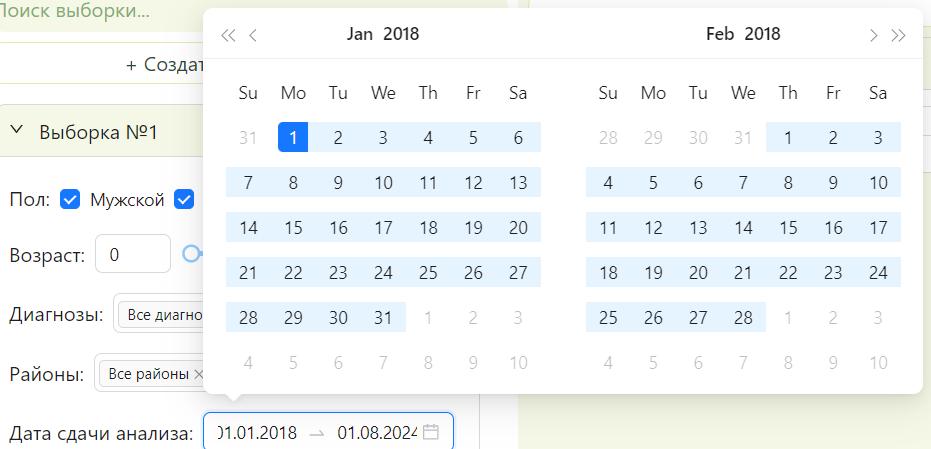
Далее предобработанные данные загружались в базу данных через чистые SQL-запросы. Для некоторых таблиц понадобилась дополнительная обработка данных.

Для возможности работы с новыми данными нужно было обновить несколько параметров при составлении выборки на клиентской части приложения. Это параметры:

1. Районы (рис. 6) – нужно было добавить район «город Тюмень»;
2. Даты исследования (рис. 7) – теперь диапазон дат исследований по умолчанию с 1 января 2018 года по 1 августа 2024 года.



*Рис. 6. Новое отображение районов*



*Рис. 7. Новое отображения дат проведения измерений*

Указанный функционал внедрён в Docker-контейнер основного приложения в модуль импорта данных.

## 3.2. Определение пола пациента по ФИО

ФИО является обязательной характеристикой для медицинских исследований, так как в датасете этот параметр не был предоставлен, было решено обучить модель классификации пола по ФИО пациента. Ход работы:

1. Предобработка данных — это важный этап в процессе анализа данных и машинного обучения, который включает в себя различные методы и техники, направленные на подготовку сырых данных к дальнейшему анализу и моделированию. В контексте работы с текстовыми данными, такими как ФИО (фамилия, имя, отчество), предобработка данных включает в себя несколько ключевых шагов, таких как нормализация текста, заполнение пробелов и разделение строк.
   1. Нормализация текста — это процесс приведения текстовых данных к единому формату, что позволяет уменьшить вариативность данных и улучшить качество анализа. Один из основных методов нормализации — это приведение строк к нижнему регистру. Это делается для того, чтобы избежать различий между строками, которые могут возникнуть из-за использования разных регистров. Например, строки "Иванов" и "иванов" будут восприниматься как разные значения, если не привести их к одному регистру. Приведение к нижнему регистру позволяет устранить такие различия и сделать анализ более точным.

Другим важным аспектом нормализации является преобразование символов в числовые значения. В данном случае используется метод, основанный на ASCII-кодах, где каждому символу соответствует уникальное числовое значение. Например, буква 'a' имеет ASCII-код 97, а 'A' — 65. Преобразование символов в числовые значения позволяет использовать текстовые данные в числовых алгоритмах, таких как машинное обучение, где требуется представление данных в числовом формате.

* 1. Заполнение пробелов — это еще один важный шаг в предобработке данных, который обеспечивает единообразие длины строк. В некоторых случаях строки могут иметь разную длину, что может вызвать проблемы при их обработке. Например, если алгоритм ожидает входные данные фиксированной длины, а строки имеют разную длину, это может привести к ошибкам. Для решения этой проблемы используется метод заполнения пробелами. В этом методе строки дополняются пробелами до заданной длины, что позволяет обеспечить одинаковую длину всех строк. Это особенно полезно при работе с нейронными сетями и другими алгоритмами, которые требуют фиксированного размера входных данных.
  2. Разделение строк — это процесс, который позволяет разбивать сложные текстовые данные на более простые и управляемые компоненты. В случае ФИО, данные могут быть представлены в разных форматах: в одной строке (например, "Иванов Иван Иванович") или в отдельных колонках (фамилия, имя, отчество). Разделение строк позволяет более эффективно обрабатывать и анализировать данные, так как каждая часть ФИО может быть обработана отдельно. Например, можно использовать метод split() для разделения строки на отдельные компоненты, что позволяет легко извлекать фамилию, имя и отчество для дальнейшего анализа или классификации. Это также упрощает процесс предобработки, так как каждая часть может быть нормализована и обработана отдельно, что повышает точность и эффективность анализа.

1. Моделирование — это ключевой этап в процессе анализа данных и машинного обучения, который включает в себя применение различных алгоритмов для создания предсказательных моделей на основе обучающих данных. Одним из популярных методов моделирования является метод ближайших соседей, известный как KNN (K-Nearest Neighbors). Этот метод используется для классификации и регрессии и основывается на принципе, что объекты, находящиеся близко друг к другу в пространстве признаков, имеют схожие характеристики.

Метод ближайших соседей (KNN) — это простой, но мощный алгоритм, который не требует предположений о распределении данных. Он работает на основе идеи, что объекты, которые находятся близко друг к другу, скорее всего, принадлежат к одному и тому же классу. Алгоритм KNN классифицирует новый объект, основываясь на классах его ближайших соседей в обучающем наборе данных.

Принцип работы KNN

1. Определение расстояния: Для начала алгоритм KNN требует определения метрики расстояния, которая будет использоваться для измерения близости между объектами. Наиболее распространенными метриками являются евклидово расстояние, манхэттенское расстояние и косинусное расстояние. Выбор метрики может существенно повлиять на результаты классификации.
2. Выбор параметра K: Параметр K определяет количество ближайших соседей, которые будут учитываться при классификации нового объекта. Например, если K=3, алгоритм будет рассматривать три ближайших соседа. Выбор значения K является критически важным, так как слишком маленькое значение может привести к переобучению (overfitting), а слишком большое — к недообучению (underfitting). Обычно K выбирается с помощью кросс-валидации, чтобы найти оптимальное значение, которое минимизирует ошибку классификации.
3. Классификация: После определения расстояний до всех объектов в обучающем наборе данных, алгоритм KNN выбирает K ближайших соседей и определяет класс нового объекта на основе голосования. Каждый сосед "голосует" за свой класс, и класс, получивший наибольшее количество голосов, становится предсказанным классом для нового объекта. В случае равенства голосов может применяться дополнительная логика, например, выбор класса с меньшим индексом [1].

Оценка модели позволяет понять, насколько хорошо модель справляется с задачей предсказания на основе данных, которые она не видела ранее. Одной из наиболее распространенных метрик для оценки качества классификационных моделей является точность (accuracy).

Точность (accuracy) — это метрика, которая измеряет долю правильных предсказаний модели относительно общего числа предсказаний. Она показывает, насколько хорошо модель классифицирует данные, и выражается в виде процента. Формально, точность можно определить следующим образом:

где:

1. (True Positives) — количество истинно положительных предсказаний (правильно предсказанные положительные классы).
2. (True Negatives) — количество истинно отрицательных предсказаний (правильно предсказанные отрицательные классы).
3. (False Positives) — количество ложноположительных предсказаний (неправильно предсказанные положительные классы).
4. (False Negatives) — количество ложноотрицательных предсказаний (неправильно предсказанные отрицательные классы).

Точность может быть полезна в различных сценариях. Например, в задаче классификации пола по ФИО, точность позволяет быстро оценить, насколько хорошо модель предсказывает пол на основе предоставленных данных. Если модель предсказывает пол с высокой точностью, это может свидетельствовать о том, что она успешно захватывает закономерности в данных.

Метрика точности имеет некоторые ограничения, которые следует учитывать:

1. Несбалансированные классы: Если в наборе данных классы сильно несбалансированы (например, 90% положительных и 10% отрицательных примеров), высокая точность может быть достигнута даже при плохом качестве предсказаний. Например, если модель всегда предсказывает положительный класс, она может достичь 90% точности, но при этом не будет правильно классифицировать ни один отрицательный класс.
2. Не учитывает типы ошибок: Точность не различает типы ошибок (ложноположительные и ложноотрицательные). В некоторых задачах, таких как медицинская диагностика, важно минимизировать количество ложноотрицательных предсказаний, так как это может привести к серьезным последствиям.

## 3.3. Анализ временных рядов

Данный модуль предназначен для анализа временных рядов медицинских данных, полученных из базы данных PostgreSQL. Основной функционал включает в себя:

1. Извлечение данных о пациентах и их тестах на основе заданных параметров (возраст, идентификатор теста).
2. Агрегацию данных по временным периодам для получения медианных значений.
3. Применение методов временного анализа, таких как экспоненциальное сглаживание и разложение временных рядов.
4. Визуализацию результатов анализа с помощью графиков.

Структура модуля включает в себя:

1. Импорт необходимых библиотек.
2. Определение параметров подключения к базе данных.
3. Функции для анализа временных рядов.
4. Примеры использования функций.

Варианты использования:

1. Прогнозирование результатов тестов на основе исторических данных.
2. Анализ трендов и сезонных паттернов в медицинских данных.
3. Визуализация данных для лучшего понимания результатов тестирования.

Выполняется SQL-запрос для получения referral\_header\_id для пациентов, соответствующих заданному возрастному диапазону и идентификатору теста. Если результаты пусты, выводится сообщение об отсутствии данных.

Если данные найдены, они извлекаются в датафрейм, который содержит информацию о дате забора образца и результатах теста.

Далее происходит преобразование столбца с датами в формат, удобный для анализа, установка этого столбца в качестве индекса, а затем определение диапазона дат для фильтрации данных, чтобы сосредоточиться на определенном временном интервале.

Потомданные группируются по заданному периоду (period\_days), и для каждой группы вычисляется медиана. Пропуски (NaN) удаляются. Если после агрегации данные пусты, выводится сообщение об отсутствии доступных данных.

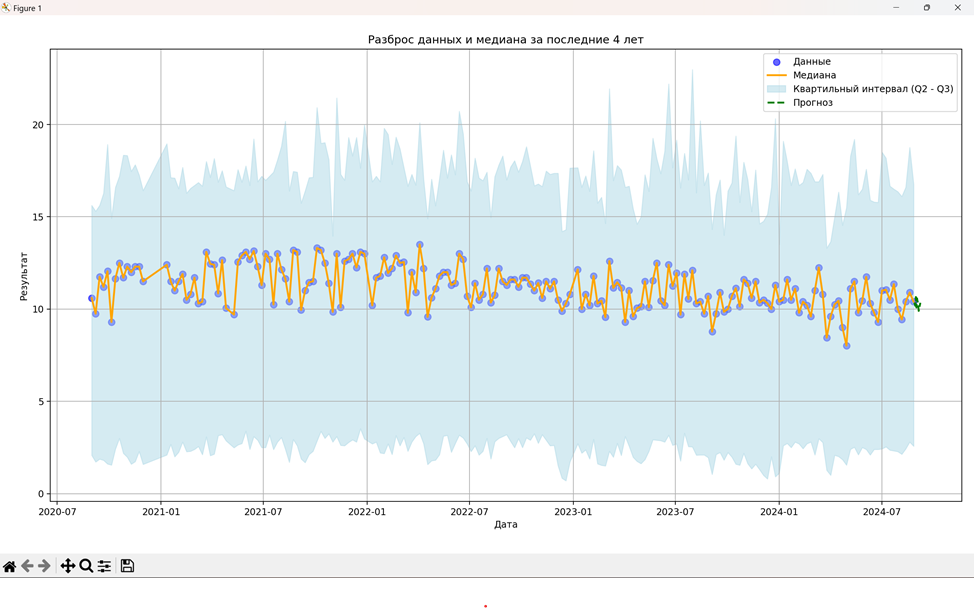
Код вычисляет 1-й (Q1) и 3-й (Q3) квартиль для сгруппированных данных, которые затем добавляются в датафрейм.

Создается модель аддитивного экспоненциального сглаживания на основе медианных значений. Модель обучается, и затем выполняется прогноз на 12 шагов вперед.

В итоге создаётся график, на котором отображаются:

1. Исходные данные (точки).
2. Линия медианы.
3. Заполнение между 2-м и 3-м квартилями.
4. Прогнозируемые значения.

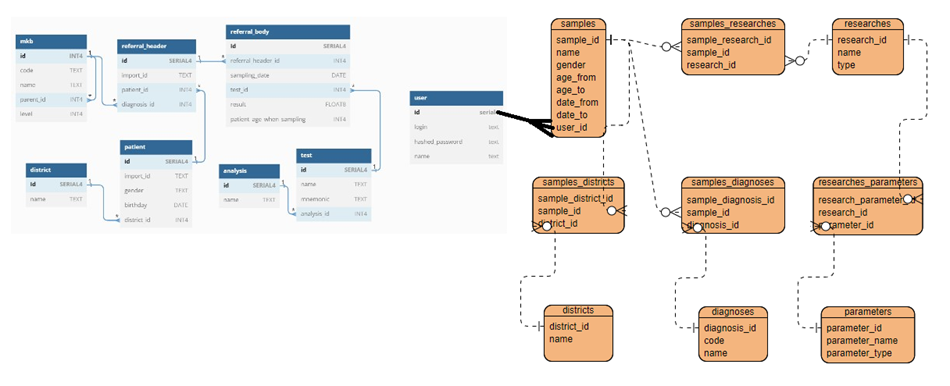
График настраивается с заголовком, подписями осей и легендой, после чего отображается (рис.8).



*Рис. 8. График разброса данных и медианы результатов измерений*

## 3.4. Сохранение параметров выборок и исследований

Для хранения выборок и исследований был рассмотрен вариант расширения существующей базы данных (рис. 9), но из-за того, что в одних и тех же полях в конструкторе выборок и исследований могли быть разные типы данных и из-за большого количества связей «многие ко многим» было решено хранить эти данные в нереляционной СУБД. В качестве такой СУБД была выбрана MongoDB, её связь с основной базой данных должна будет проходить через серверную часть приложения.



*Рис. 9. Структура реляционной БД при хранении выборок и исследований*

Предполагалось, что связь между пользователем и его выборками будет происходить по ID пользователя в основной (реляционной) базе данных, поле с ID пользователя должно было стать частью структуры документа выборки.

Параметры выборок и исследований должны были иметь такую же связь, в документах исследования будет поле с ID выборок, которые использовались для его проведения.

Была изучена теория, но реализовать запланированное в приложении не получилось, так как возникли проблемы с локальным развёртыванием текущей версии и пониманием унаследованного кода.

## 3.5. Тестирование

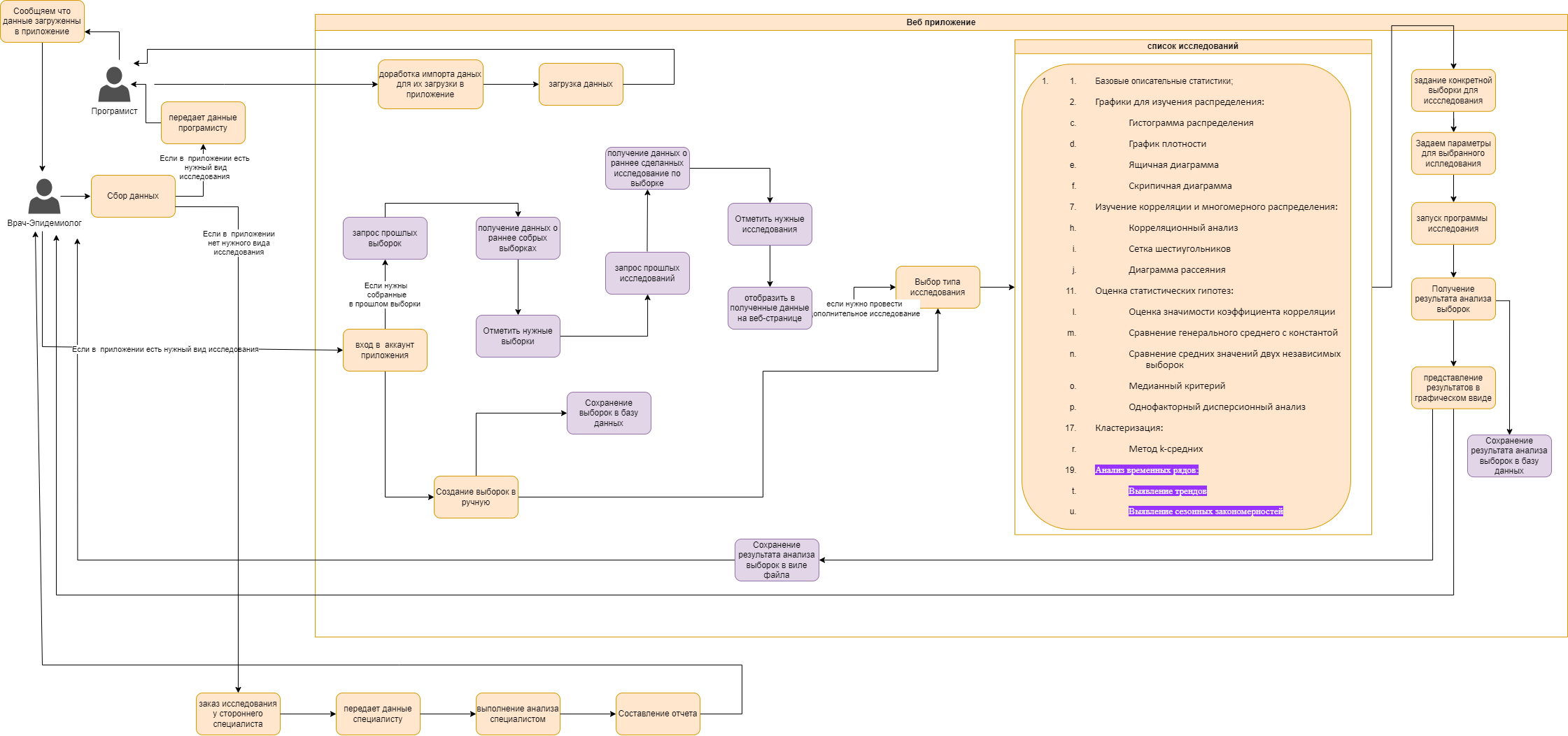
Так как большинство функций имеет чётко заданную структуру и типизацию входных значений, тестирование в основном было бинарным, функция работает или не работает, график отображается или не отображается.

Адекватность графиков временных рядов участники команды проверить не могут из-за отсутствия квалификации в сфере медицины. Этот функционал будет проверять сам заказчик.

## 3.6. Добавленный функционал

На момент составления отчёта держатель приложения тестирует функционал загрузки новых данных в базу данных, в виде отдельного модуля реализованы функции для анализа временных рядов - сглаживание, выделение тренда и сезонных закономерностей, эти функции находятся на этапе встраивания в основное приложения. Изучены теоретические и логические основы сохранения параметров выборок и исследования, разработаны макеты интерфейса для соответствующих страниц.

Запланированный функционал можно просмотреть на рис.10.



*Рис. 10. Функциональная схема проекта. Фиолетовый - планируемый добавленный функционал*

# ЗАКЛЮЧЕНИЕ

В рамках проекта были выполнены не все поставленные задачи. Не закончено внедрение функций анализа временных рядов, изучена только теоретическая часть сохранения параметров выборок и исследований, формирование отчётов не было начато.

Однако полностью реализована загрузка новых данных в базу данных, включая модель классификации пола пациента по его ФИО. Эти данные в скором времени можно будет использовать во всех уже реализованных методах статистической обработки, после тестирования расширения для модуля импорта данных держателем приложения.

Причина неудачи в выполнении полного объёма работ заключается в том, что команде не сразу удалось наладить контакт с заказчиком и держателем приложения. Также из-за некорректного расчёта времени и общей нагрузки не удалось достичь всех поставленных целей в установленный срок.

# СПИСОК ЛИТЕРАТУРЫ

1. Брюс П., Брюс, Э., Гедек П. Практическая статистика для специалистов Data Science. Санкт-Петербург, 2021. 352 с. URL: https://batrachos.com/sites/default/files/pictures/Books/Bruce\_Bruce\_2018\_Practical%20Statistics%20for%20Data%20Scientists.pdf (дата обращения: 15.12.2024).
2. Петри А., Сэбин К. Наглядная медицинская статистика. Москва: ГЭОТАР-медиа, 2009. 168 с. URL: https://djvu.online/file/9wRpIwFLrBSHx?ysclid=m32vyqlwrk993801704 (дата обращения: 15.12.2024).
3. Наркевич А.Н., Виноградов К.А. Выбор метода для статистического анализа медицинских данных и способа графического представления результатов // Социальные аспекты здоровья населения. 2019. No4 (68). URL: http://vestnik.mednet.ru/content/view/1092/30/lang,ru/ (дата обращения: 15.12.2024).
4. Гланц С. Медико-биологическая статистика. Москва: Практика, 1999. 462 с. URL: https://medstatistic.ru/articles/glantz.pdf (дата обращения: 15.12.2024).
5. Федоров Н., Понизова В. Статистический анализ временных рядов. Санкт-Петербург: СПБГУ, 2020. 79 с. URL: https://www.statmod.ru/wiki/\_media/study:spring2020:ts:notes20200406.pdf (дата обращения: 15.12.2024).