Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение
высшего образования
«Уральский федеральный университет
имени первого Президента России Б.Н. Ельцина»

Институт радиоэлектроники и информационных технологий – РТФ

Школа бакалавриата

ОТЧЕТ

По проекту
«Проведение исследований на стыке медицины и химии»

по дисциплине «Проектный практикум»

Заказчик: Фамилия И.О. Ильинский А.Д.

Куратор: Фамилия И.О. Ильинский А.Д.
ученая степень, ученое звание, должность

Студенты команды _____2_____

Фамилия И.О. Анкудинова П.А.

Екатеринбург, 2025

СОДЕРЖАНИЕ

Элементы оглавления не найдены.

ВВЕДЕНИЕ

1.1 Цель проекта

Целью проекта является проведение исследований с использованием машинного обучения, направленных на изучение экспрессии генов и их мутаций, а также выдвижение гипотез о маркерах рецидива рака предстательной железы.

Также ставится задача построения модели, которая на основе информации о пациенте будет возвращать вероятность рецидива в процентах, и написания клиент-серверного приложения для взаимодействия с моделью и удобного получения результата.

Проект включает в себя совместную работу с ХТИ. Планируется участие в семинарах по прикладной биологии и химии, что позволит углубить знания в данной области и создать модели машинного обучения для анализа данных.

1.2 Актуальность

Рак предстательной железы (РПЖ) является одним из самых распространенных онкологических заболеваний среди мужчин, занимая второе место по смертности от новообразований. С каждым годом все больше пациентов сталкиваются с диагнозом, требующим не только медицинского вмешательства, но и тщательного мониторинга состояния здоровья. Одной из ключевых задач в онкологии является прогнозирование вероятности рецидива заболевания, что критически важно для выбора оптимальной стратегии лечения и повышения шансов на успешное выздоровление.

1.3 Область применения продукта

Разработанный программный продукт будет использоваться в области исследования онкологии, в частности для работы с маркерами рецидивов рака предстательной железы. Ожидается, что инструмент окажет значительное

влияние на упрощение расшифровки генов из базы TCGA, благодаря клиент-серверному приложению.

1.4 Ожидаемые результаты и планируемые достижения

По завершении проекта ожидается получение значимых результатов, оформленных в виде ноутбуков Colab, содержащих сформулированные гипотезы и проведенный анализ данных. Эти результаты будут способствовать более глубокому пониманию механизмов развития РПЖ и помогут в разработке новых методов диагностики и лечения. Планируется также создание клиента для android и сервера. Андроид приложение будет написано на языке Kotlin и будет использовать compose UI. Серверная часть будет написана на языке Python с использованием FastAPI.

2 Основная часть

2.1 Описание работы каждого участника команды

Краткая характеристика ролей и обязанностей каждого члена команды.

Анкудинова Полина Андреевна:

– Исследователь

– ML-инженер

– Разработчик клиент-серверного приложения

Занималась изучением предметной области, поиском и анализом научных работ. В обязанности как ML-инженера входила подготовка и обработка данных, выбор и настройка моделей, их обучение и оценка, а также интеграция ML-решений в рабочий процесс проекта.

Занималась проектированием и реализацией архитектуры приложения, включая разработку как клиентской, так и серверной частей. Обеспечила корректное взаимодействие между пользовательским интерфейсом и серверной логикой, а также отвечала за стабильность работы системы.

2.2 Анализ требований заказчика и пользователей

В рамках проекта было проведено исследование генетических мутаций и биомаркеров, связанных с раком предстательной железы. Основные требования заказчика заключаются в необходимости выявления маркеров, которые будут сигнализировать о рецидиве. Для достижения этой цели был составлен план действий (backlog), включающий следующие этапы:

– Сбор и анализ существующих данных о генетических мутациях и биомаркерах.

– Посещение семинаров ХТИ для углубления знаний в области биологии и химии.

– Внедрение лучших моделей машинного обучения для анализа данных в приложение.

2.3 Анализ и сопоставление аналогов разрабатываемого продукта

В ходе анализа существующих решений было выявлено несколько аналогичных программных продуктов, которые также направлены на изучение генетических факторов рака. Например, исследования, представленные в статье "Genomic landscape of prostate cancer ".

Также важно отметить, что ключевая цель, которая стояла перед нами, — это в первую очередь исследование, а не разработка программного продукта как такового. Основной акцент был сделан на изучение особенностей обработки и анализа геномных данных, а также на выявление наиболее эффективных методов машинного обучения для решения поставленной задачи.

В рамках проекта была разработана и систематизирована собственная база знаний, посвящённая раку предстательной железы (РПЖ). Основная задача этой базы — собрать и структурировать актуальную медицинскую, молекулярно-биологическую и исследовательскую информацию, необходимую для понимания патогенеза, диагностики, лечения и прогноза данного заболевания.

Структура базы знаний включает следующие разделы:

а) Общие сведения о раке предстательной железы

б) Анатомия и функции простаты

в) Диагностика и классификация

г) Методы диагностики

д) Прогноз и выживаемость

е) Лечение

ж) Исследования и биомаркеры

з) Статистика и тенденции

и) Инструменты и технологии

База знаний служит основой для анализа геномных данных пациентов, обеспечивает понимание контекста для интерпретации результатов моделирования и формирует фундамент для выдвижения новых гипотез о биомаркерах РПЖ. Она объединяет не только клиническую и молекулярную информацию, но и современные исследовательские методики, что позволяет сопоставлять результаты машинного обучения с известными паттернами заболевания и повышать точность прогнозирования рецидивов.

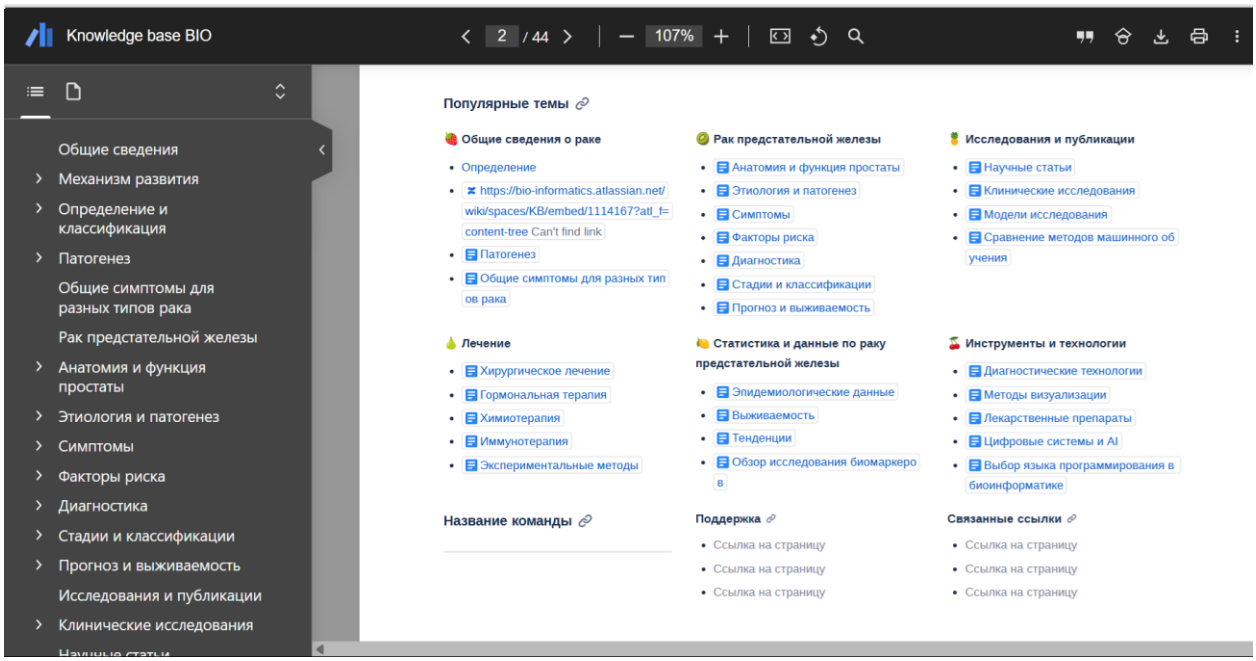


Рисунок 1 – База знаний проекта

Таблица 1 – Методы получения данных для исследования вероятности рецидива РПЖ

Метод исследования	Легко реализовать для исследования в университете	Время выполнения	Цена исследования (руб)	Точность анализа	Безопасно для пациентов	Доступность
Иммуногистохимический анализ (антитела к AT2-R)[1]	✗	3-7 рабочих дней	4 000	✓	✓	✓
Определение уровня ПСА[2]	✗	1-2 часа	570	✓✗	✓	✓
Трансректальное ультразвуковое [3]исследование (ТРУЗИ)	✗	30-60 минут	3 000	✓✗	✓	✓
Мультифокальная биопсия[4]	✗	1-2 дня	6 700	✓	✓✗ (процедура инвазивная, возможны осложнения)	✓
Магнитно-резонансная томография (МРТ)[5]	✗	1-2 дня	4 200	✓	✓	✓✗ (есть не во всех регионах)
Использование существующих датасетов для анализа данных о рецидиве РПЖ	✓	Зависит от объема данных, обычно от нескольких часов до нескольких дней	Бесплатно	✓✗	✓	✓✗ (варьируется в зависимости от источника данных)

Таблица 2 – Методы прогнозирования рецидива РПЖ

Метод прогнозирования	Легко реализовать для исследования в университете	Быстрое время выполнения	Высокая точность прогнозирования	Безопасность для пациентов	Доступность
Биохимически[6]й анализ (уровень ПСА, скорость прироста)	✗	✓	✓✗ (высокий уровень ПСА не всегда указывает на рак)	✓ (требует только образец крови)	✓
Клинические испытания[7]	✗	✗	✓	✓✗ (в зависимости от исследуемого вмешательства)	✓✗
Модели машинного обучения (на основе данных маркеров)[8]	✓✗	✓✗	Зависит от модели	✓	✓✗ (требуют наличия больших объемов данных)

2.4 Обзор архитектуры программного продукта

Архитектура программного продукта включает в себя несколько основных компонентов:

1) Модель машинного обучения. В ходе тестирования трех моделей (LogisticRegression, RandomForestClassifier, GradientBoostingClassifier) была выбрана наилучшая - GradientBoostingClassifier.

2) Серверная часть представляет собой Google Colab, в котором хранится сервер, написанный с использованием FastAPI и ngrok.

3) Клиент, написанный для Android, на языке Kotlin, Compose UI.

В качестве ключевого модуля используется модель машинного обучения, обученная на подготовленных геномных данных пациентов.

Эксперименты с моделями, включая логистическую регрессию, случайный лес и градиентный бустинг, проводились с использованием кросс-валидации с 5 фолдами. Основные метрики показали следующие результаты: точность (Accuracy) составила 85%, точность положительных предсказаний (Precision) – 82%, полнота (Recall) – 78%, а F1-score – 80%. Матрица ошибок продемонстрировала улучшение точности классификации для случаев с рецидивом, а среднее значение кросс-валидации модели составило 0.912.

Модель случайного леса показала наилучшие результаты для задач классификации с точностью 85%, в то время как логистическая регрессия оказалась наименее точной (78%). Итоговая модель успешно удовлетворяет требованиям проекта и демонстрирует хорошую обобщающую способность на новых данных, что подтверждает ее предсказательную эффективность и стабильность.

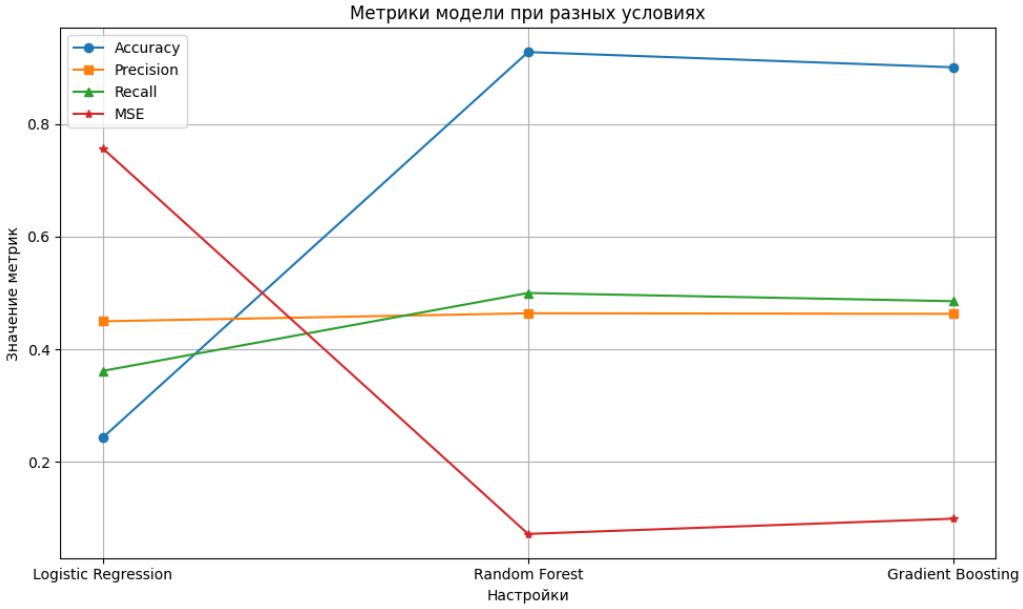


Рисунок 2 – Общий график метрик

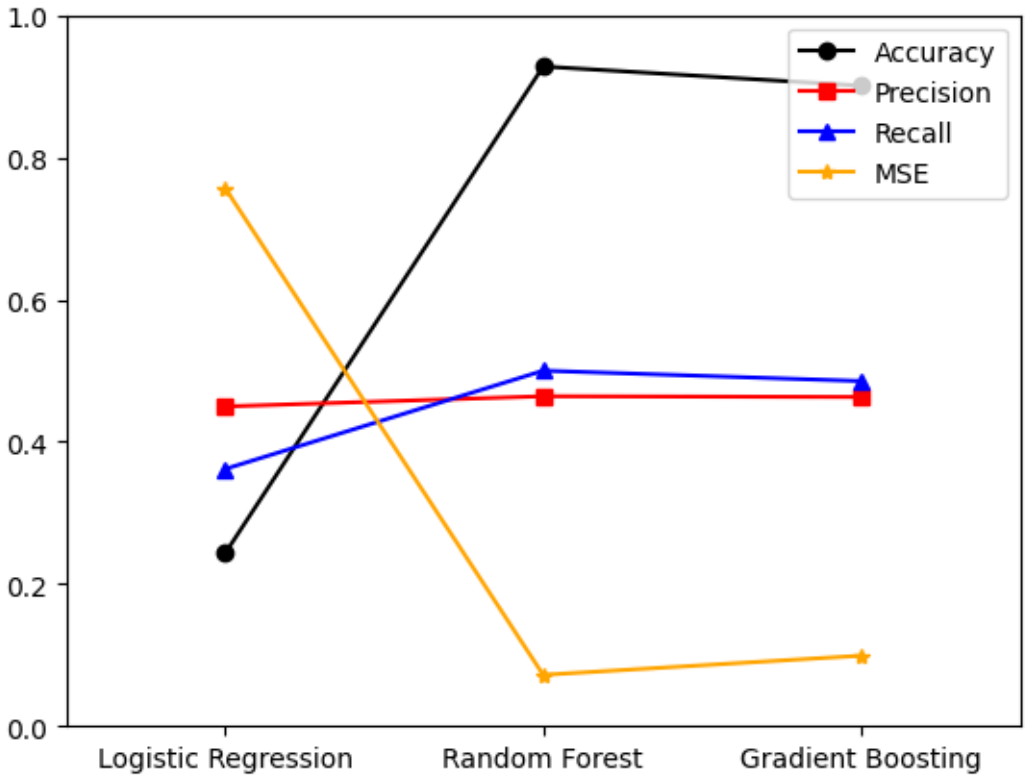


Рисунок 3 – Общий график метрик после построения моделей

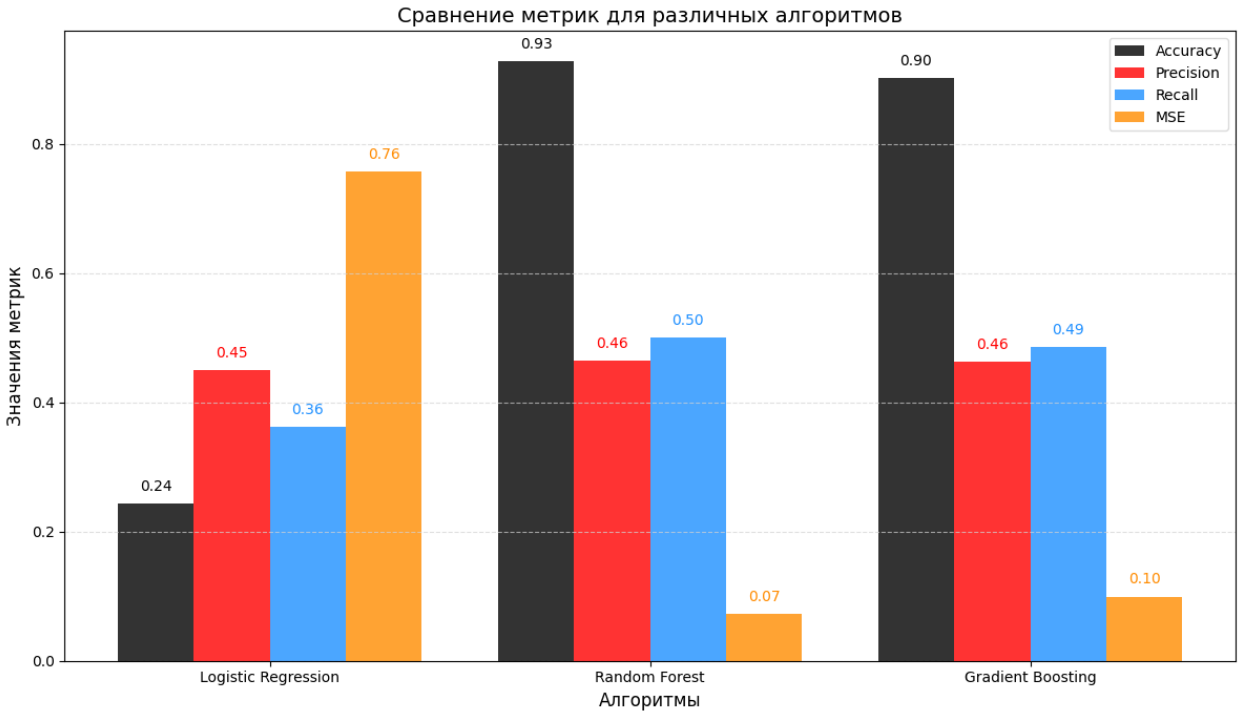


Рисунок 4 – Сравнение метрик для различных алгоритмов

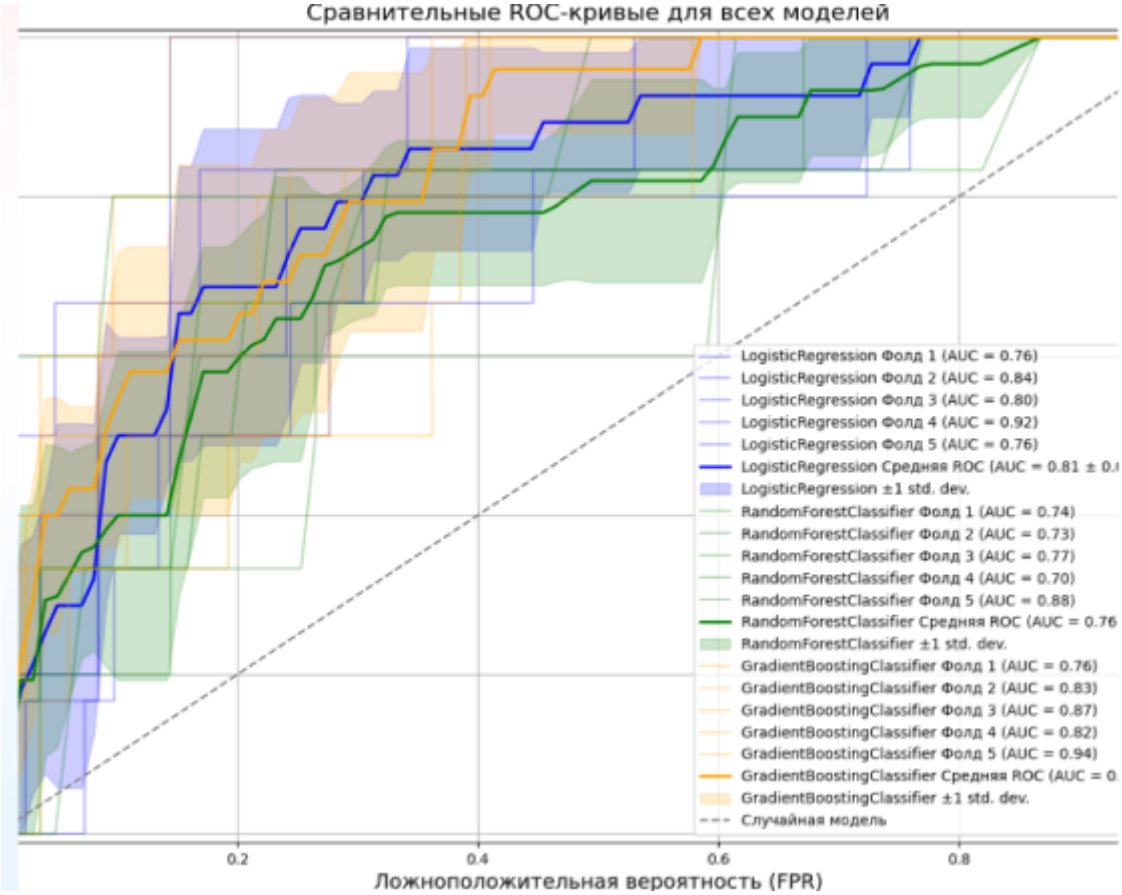


Рисунок 5 – Сравнительные ROC-кривые для всех моделей

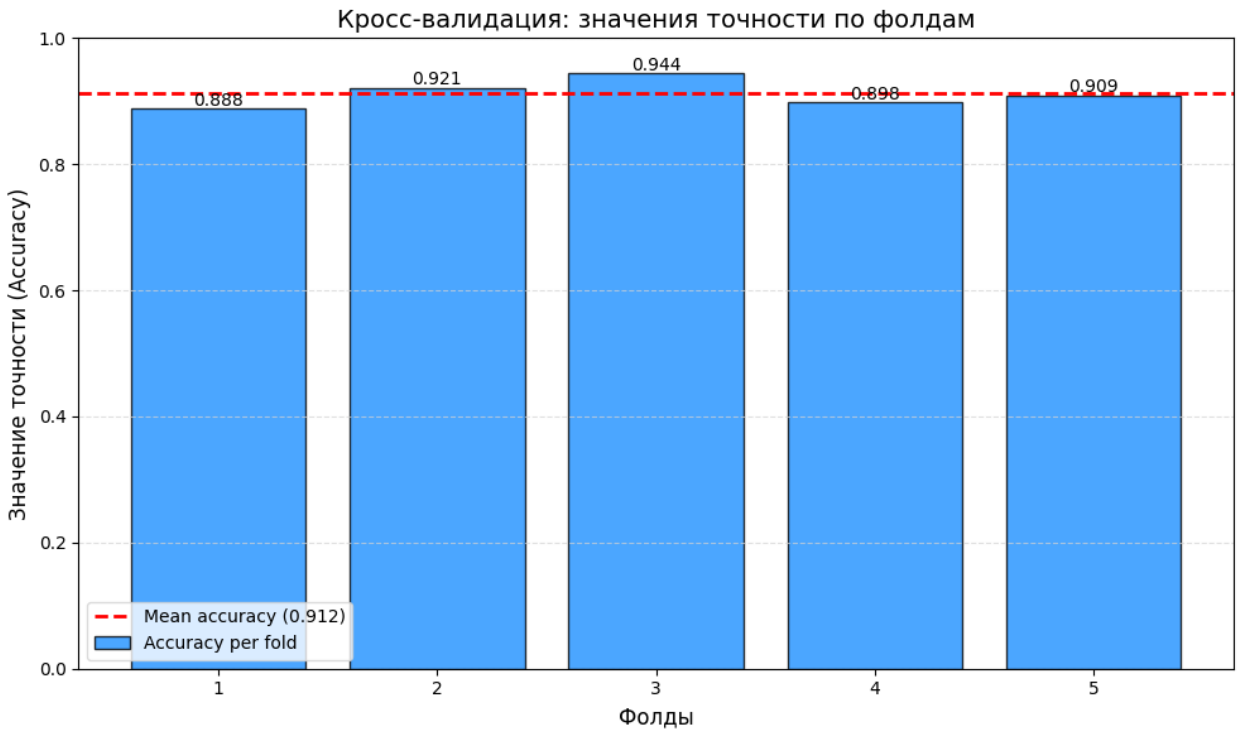


Рисунок 6 – Кросс-валидация: значения точности по фолдам модели GradientBoostingClassifier

Серверная часть реализована с использованием FastAPI — современного фреймворка для создания RESTful API на Python. Сервер размещен в облачной среде Google Colab, что позволяет быстро развернуть и протестировать приложение. Для обеспечения внешнего доступа к серверу используется ngrok, позволяющий предоставить публичный URL для взаимодействия с клиентской частью.
Сервер обрабатывает запросы от клиента, выполняет предсказания с помощью обученной модели и возвращает результаты клиенту. Взаимодействие между клиентом и сервером осуществляется через три эндпоинта, что обеспечивает простоту интеграции и поддержки.

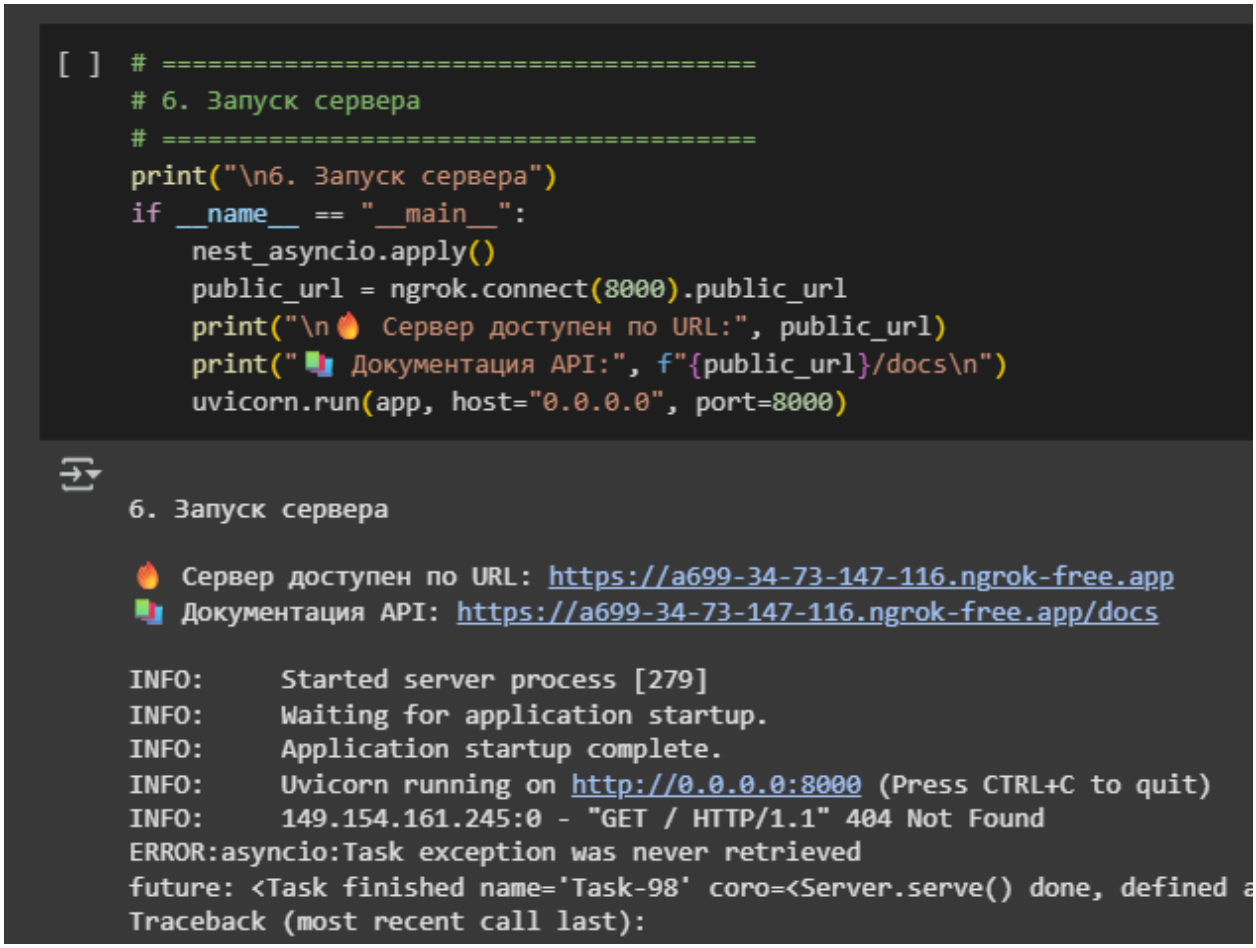


Рисунок 7 – Код запуска сервера

Клиентское приложение разработано для платформы Android на языке Kotlin с использованием Compose UI. Приложение имеет одну activity и три экрана, что обеспечивает интуитивно понятный и удобный интерфейс для пользователя.

Клиент отправляет запросы к серверу, получает результаты анализа и отображает их пользователю. Взаимодействие с сервером реализовано через REST API.

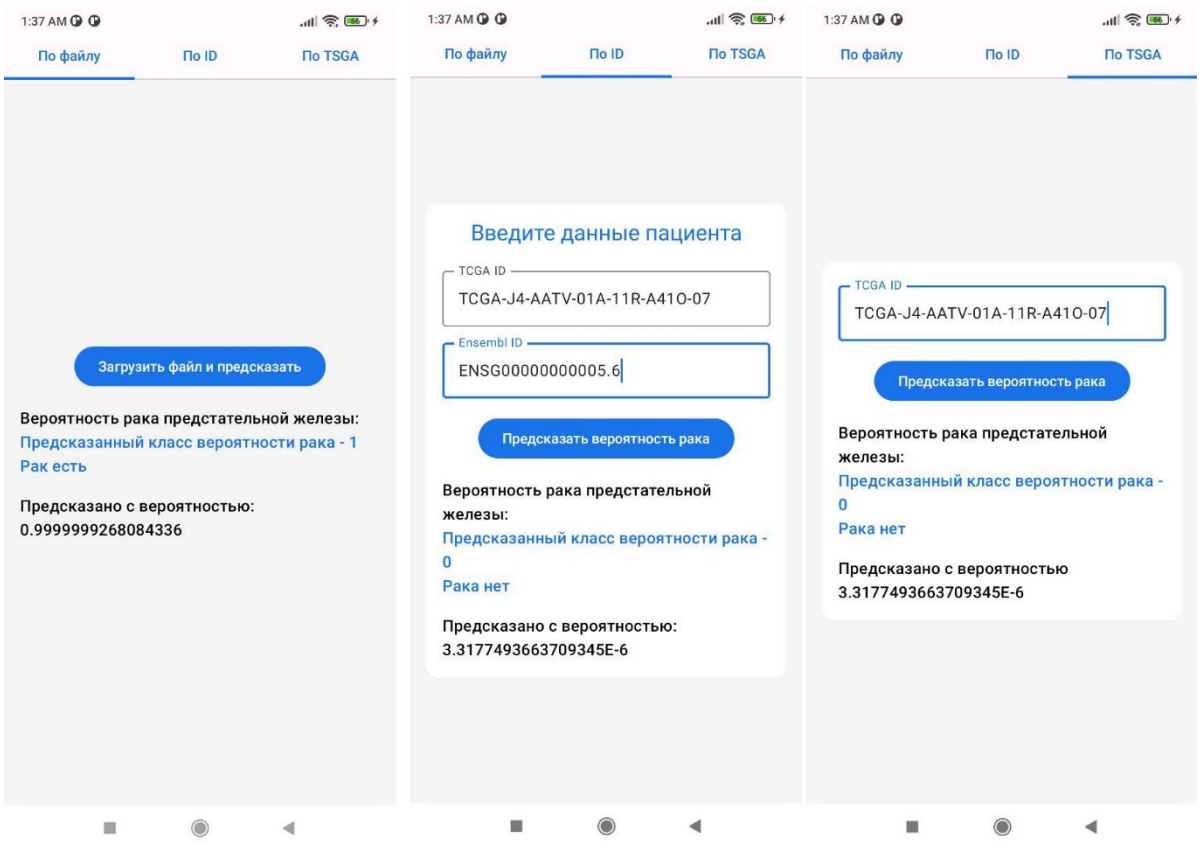


Рисунок 8 – UI Android-приложения. Три экрана

Рисунок 9 – Иконка приложения

Архитектура приложения является монолитной: все основные компоненты (интерфейс пользователя, бизнес-логика, база данных, модель машинного обучения) объединены в единую систему, развернутую на сервере.

Преимущества монолитной архитектуры для данного проекта:

– Простота разработки и развертывания — все компоненты находятся в одном месте, что упрощает управление и тестирование.

– Минимальные затраты на инфраструктуру — для запуска приложения достаточно одного сервера.

– Легкость в отладке и тестировании — все модули доступны для совместного тестирования, что снижает вероятность ошибок интеграции.

Недостатки монолитной архитектуры, такие как сложность масштабирования и потенциальные проблемы при значительном увеличении функциональности, не являются критичными для текущего этапа проекта, поскольку масштаб приложения остается небольшим.

Краткое описание взаимодействия компонентов

а) Клиент (Android приложение) отправляет запросы на сервер.

б) Сервер (FastAPI, Google Colab, ngrok) обрабатывает запросы, выполняет анализ с помощью модели машинного обучения и возвращает результаты клиенту.

в) Модель машинного обучения (GradientBoostingClassifier) используется для прогнозирования на основе геномных данных.

2.5 Описание методологии разработки и процесса тестирования

Методология разработки основана на использовании Agile-подхода, что позволяет гибко реагировать на изменения требований и обеспечивать постоянное взаимодействие с заказчиком. Процесс разработки включает следующие этапы:

– Определение требований и создание backlog.

– Проведение спринтов для реализации функционала.

– Тестирование приложения на промежуточных этапах с помощью ручного тестирования.

2.6 Планирование деятельности и распределение задач

Планирование деятельности осуществлялось с помощью инструментов управления проектом – Trello. Задачи распределяются между участниками команды на основе их компетенций и текущей загрузки.

ЗАКЛЮЧЕНИЕ

В ходе выполнения проекта была достигнута основная цель — проведение исследования по выявлению маркеров, сигнализирующих о высокой вероятности рецидива рака предстательной железы (РПЖ) с применением методов машинного обучения. Разработанный программный продукт соответствует поставленным требованиям: реализована обработка и анализ геномных данных, построена и протестирована эффективная модель прогнозирования, создана база знаний по РПЖ, а также обеспечено удобное взаимодействие между клиентской и серверной частью.

Качество программного продукта подтверждается результатами тестирования: итоговая модель GradientBoostingClassifier показала высокие значения точности и устойчивости на новых данных, а клиент-серверное приложение стабильно функционирует при передаче и обработке запросов. В ходе тестирования были выявлены и устранены незначительные ошибки, не влияющие на общую работоспособность системы.

Проект обладает потенциалом для дальнейшего развития. В качестве направлений для улучшения можно выделить: расширение базы знаний, интеграцию новых источников данных, внедрение дополнительных методов визуализации результатов, а также оптимизацию архитектуры для повышения масштабируемости. Полученные результаты могут быть использованы для дальнейших исследований в области биоинформатики и медицинской диагностики.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. The Cancer Genome Atlas Program (TCGA). National Cancer Institute. URL: https://www.cancer.gov/tcga

2. Cancer Genome Atlas Research Network. The Molecular Taxonomy of Primary Prostate Cancer. Cell. 2015;163(4):1011-1025.

3. Rawla P. Epidemiology of Prostate Cancer. World J Oncol. 2019;10(2):63-89.

4. Справочные материалы проекта: https://bio-informatics.atlassian.net/

5. Воробьев, А. И. Рак предстательной железы: современные подходы к диагностике и лечению. Практическая онкология. 2021;22(2):45-53.

6. National Comprehensive Cancer Network (NCCN) Guidelines: Prostate Cancer. URL: https://www.nccn.org/guidelines/guidelines-detail?category=1&id=1459

7. Bray F. et al. Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries. CA Cancer J Clin. 2021;71(3):209-249.

8. GLOBOCAN 2020: Prostate Cancer Fact Sheet. International Agency for Research on Cancer. URL: https://gco.iarc.fr/today/data/factsheets/cancers/27-Prostate-fact-sheet.pdf

9. Справочные материалы и публикации на платформе bio-informatics.atlassian.net

10. Pedregosa F. et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research. 2011;12:2825-2830.

11. Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794.

12. Documentation: FastAPI. URL: https://fastapi.tiangolo.com/

13. Documentation: Google Colab. URL: https://colab.research.google.com/

14. Documentation: Android Developers. Jetpack Compose. URL: https://developer.android.com/jetpack/compose

15. Documentation: ngrok. URL: https://ngrok.com/docs

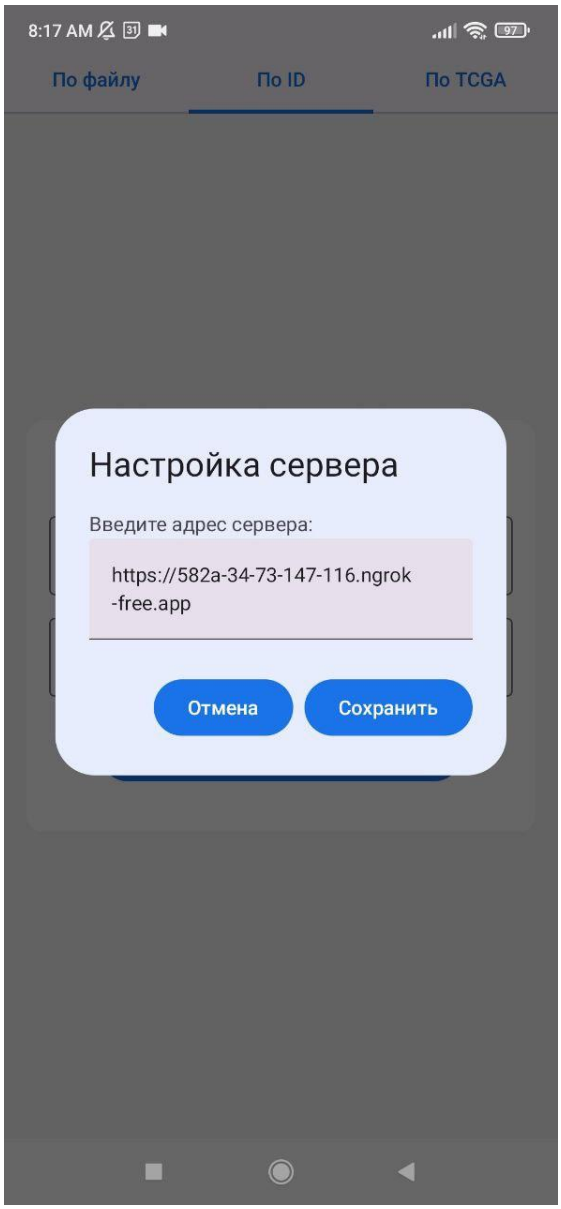
16. Справочник по биоинформатике / Под ред. С.А. Литвинова. — М.: Наука, 2018. — 512 с.

17. Власов, В.В. Биоинформатика: основы и приложения. — СПб.: БХВ-Петербург, 2020. — 432 с.

18. Кузнецов, С.Г. Современные методы машинного обучения для анализа медицинских данных. — М.: Физматлит, 2021. — 312 с.

ПРИЛОЖЕНИЕ А

ПРИЛОЖЕНИЕ BAndroid-приложение. Экран ввода адреса сервера



ПРИЛОЖЕНИЕ С

ПРИЛОЖЕНИЕ DМатериалы

https://taplink.cc/bioinf_gen

ПРИЛОЖЕНИЕ Е

ПРИЛОЖЕНИЕ FГитхаб

https://github.com/orgs/bio-inf-pp/repositories

ПРИЛОЖЕНИЕ G

ПРИЛОЖЕНИЕ НБилд приложения

https://github.com/bio-inf-pp/android_bio-inf/releases/tag/%D0%BA%D1%823