МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное

учреждение высшего образования

«Казанский национальный исследовательский технический

университет им. А.Н. Туполева – КАИ»

(КНИТУ – КАИ)

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

по дополнительной профессиональной программе профессиональной переподготовки

«Data science. Искусственный интеллект»

по теме: «БИНАРНАЯ КЛАССИФИКАЦИЯ ДЛЯ ПРЕДИКТИВНОГО ОБСЛУЖИВАНИЯ ОБОРУДОВАНИЯ»

|  |
| --- |
| **Работу выполнил:**  Ишгулов Ратмир Русланович |
|  |
| **Руководитель выпускной квалификационной работы:**  к.т.н, доцент Смирнова Гульнара Сергеевна |
|  |
|  |

Казань - 2025 г.

Оглавление

[ВВЕДЕНИЕ 3](#_Toc188126877)

[ОСНОВНАЯ ЧАСТЬ 4](#_Toc188126878)

[Описание датасета 4](#_Toc188126879)

[Предобработка данных 6](#_Toc188126880)

[Разделение данных 8](#_Toc188126881)

[Оценка модели 10](#_Toc188126883)

[Обучение моделей 13](#_Toc188126882)

Обоснование выбора [Streamlit-приложение 17](#_Toc188126884)

[ЗАКЛЮЧЕНИЕ 18](#_Toc188126885)

[СПИСОК ЛИТЕРАТУРЫ 20](#_Toc188126886)

ВВЕДЕНИЕ

Данный проект посвящен разработке модели машинного обучения для решения задачи предиктивного обслуживания оборудования. Основная цель проекта — создать систему, способную предсказывать вероятность отказа оборудования (Target = 1) или его отсутствие (Target = 0) на основе различных параметров работы.

Предиктивное обслуживание является крайне актуальной задачей в современной промышленности. В отличие от планового обслуживания, при котором регламентные работы проводятся через фиксированные промежутки времени независимо от состояния оборудования, предиктивное обслуживание основывается на фактическом состоянии оборудования и прогнозировании возможных поломок до их возникновения. Это позволяет минимизировать время простоя оборудования, снизить затраты на техническое обслуживание, увеличить срок службы оборудования и повысить безопасность производства.

В рамках проекта была разработана модель бинарной классификации, которая анализирует данные о различных параметрах работы оборудования (температура, скорость вращения, крутящий момент, износ инструмента и др.) и предсказывает вероятность его отказа. Результаты работы представлены в виде интерактивного Streamlit-приложения с возможностью загрузки данных, обучения и оценки моделей, а также выполнения предсказаний.

Ссылка на git-репозиторий с проектом:

<https://github.com/RatmirTech/KAI.PrediMaint>

ОСНОВНАЯ ЧАСТЬ

Описание датасета.

В данном проекте используется датасет "AI4I 2020 Predictive Maintenance Dataset" из репозитория UCI Machine Learning.

Источник данных:

<https://archive.ics.uci.edu/dataset/601/predictive+maintenance+dataset>

Датасет содержит синтетические данные, моделирующие работу промышленного оборудования. Набор данных состоит из 10,000 записей с 14 признаками.

Описание признаков и целевой переменной:

1. Признаки:

- UDI: Уникальный идентификатор записи.

- Product ID: Идентификатор продукта (L, M, H) и серийный номер.

- Type: Тип продукта (L, M, H).

- Air temperature [K]: Температура окружающей среды в Кельвинах.

- Process temperature [K]: Рабочая температура процесса в Кельвинах.

- Rotational speed [rpm]: Скорость вращения в оборотах в минуту.

- Torque [Nm]: Крутящий момент в Ньютон-метрах.

- Tool wear [min]: Износ инструмента в минутах.

1. Целевые переменные:

- Machine failure: Бинарная метка (1 — отказ оборудования, 0 — отказ не произошел).

- TWF: Отказ из-за износа инструмента (Tool Wear Failure).

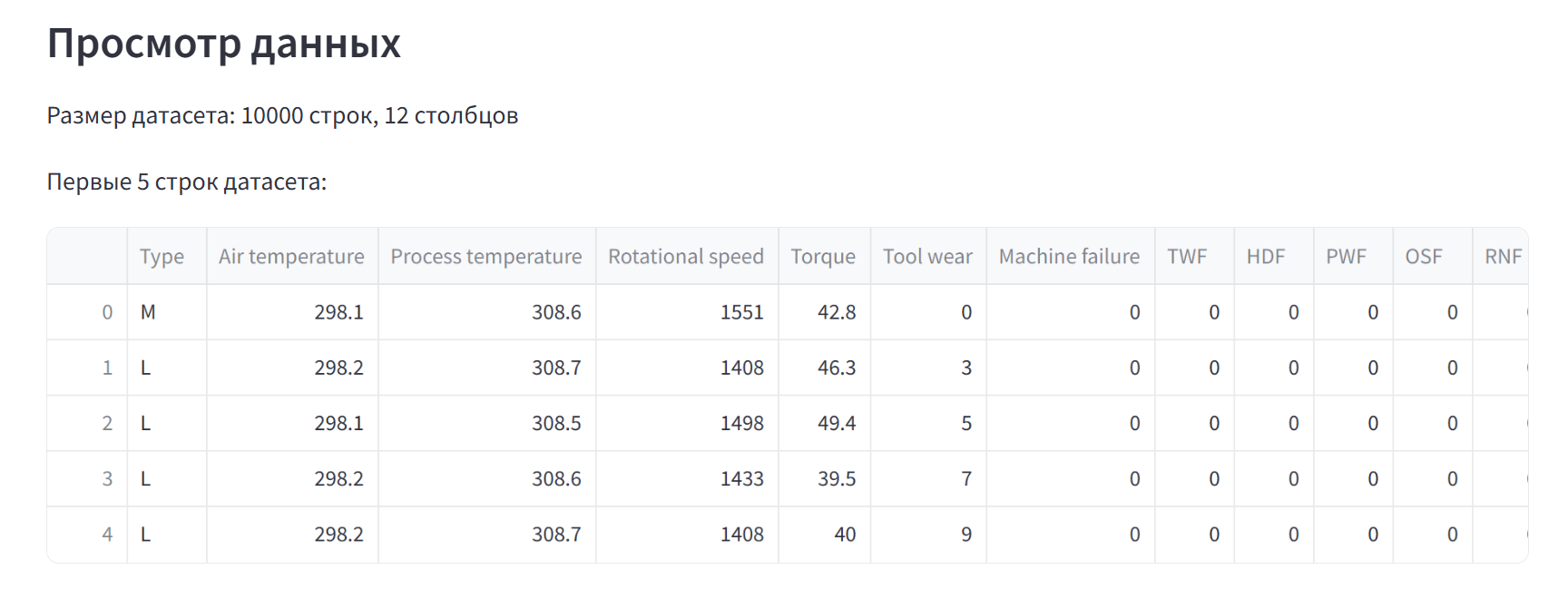
- HDF: Отказ из-за недостаточного теплоотвода (Heat Dissipation Failure).

- PWF: Отказ из-за недостаточной или избыточной мощности (Power Failure).

- OSF: Отказ из-за перегрузки (Overstrain Failure).

- RNF: Случайный отказ (Random Failure).

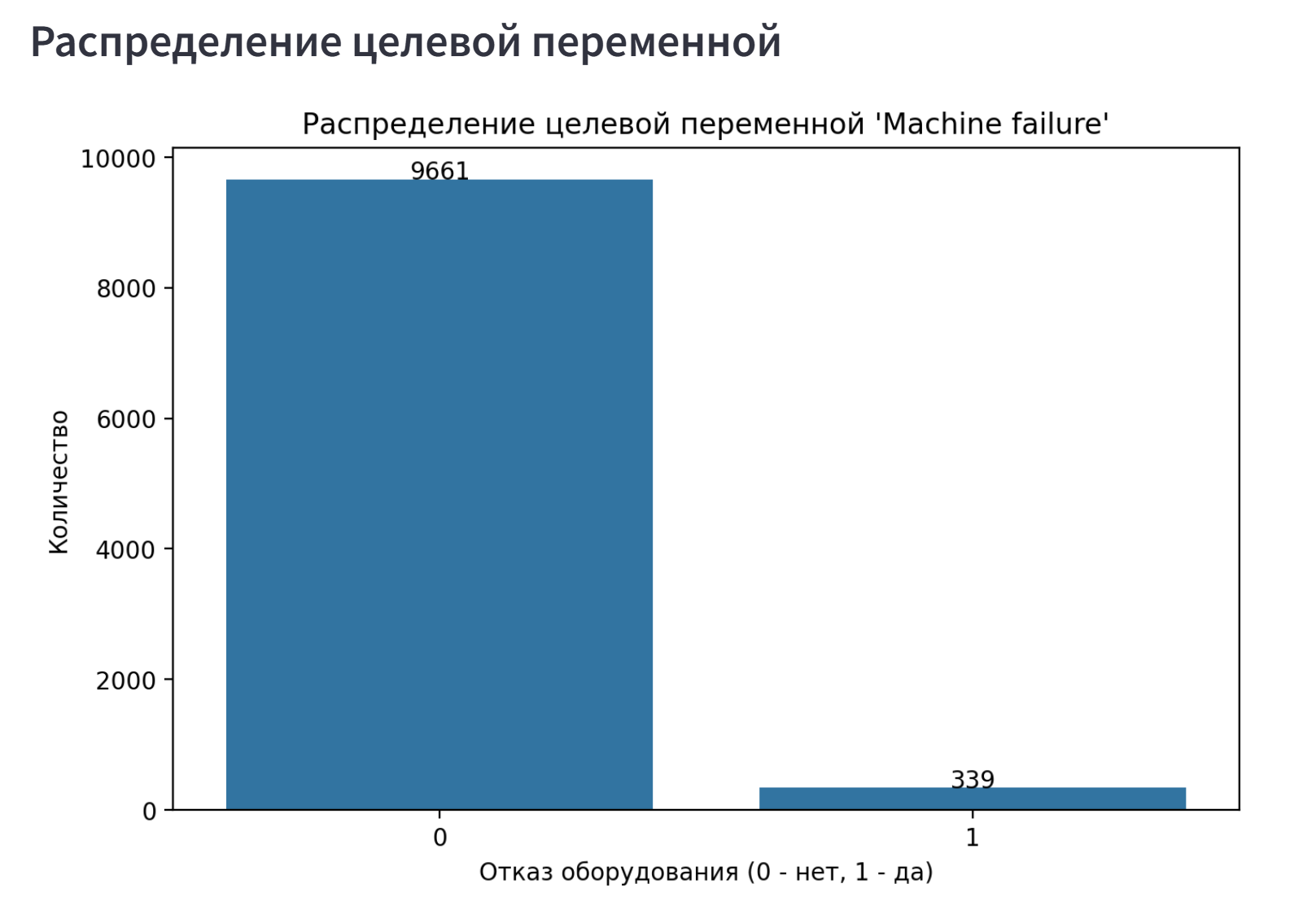
Примеры данных:



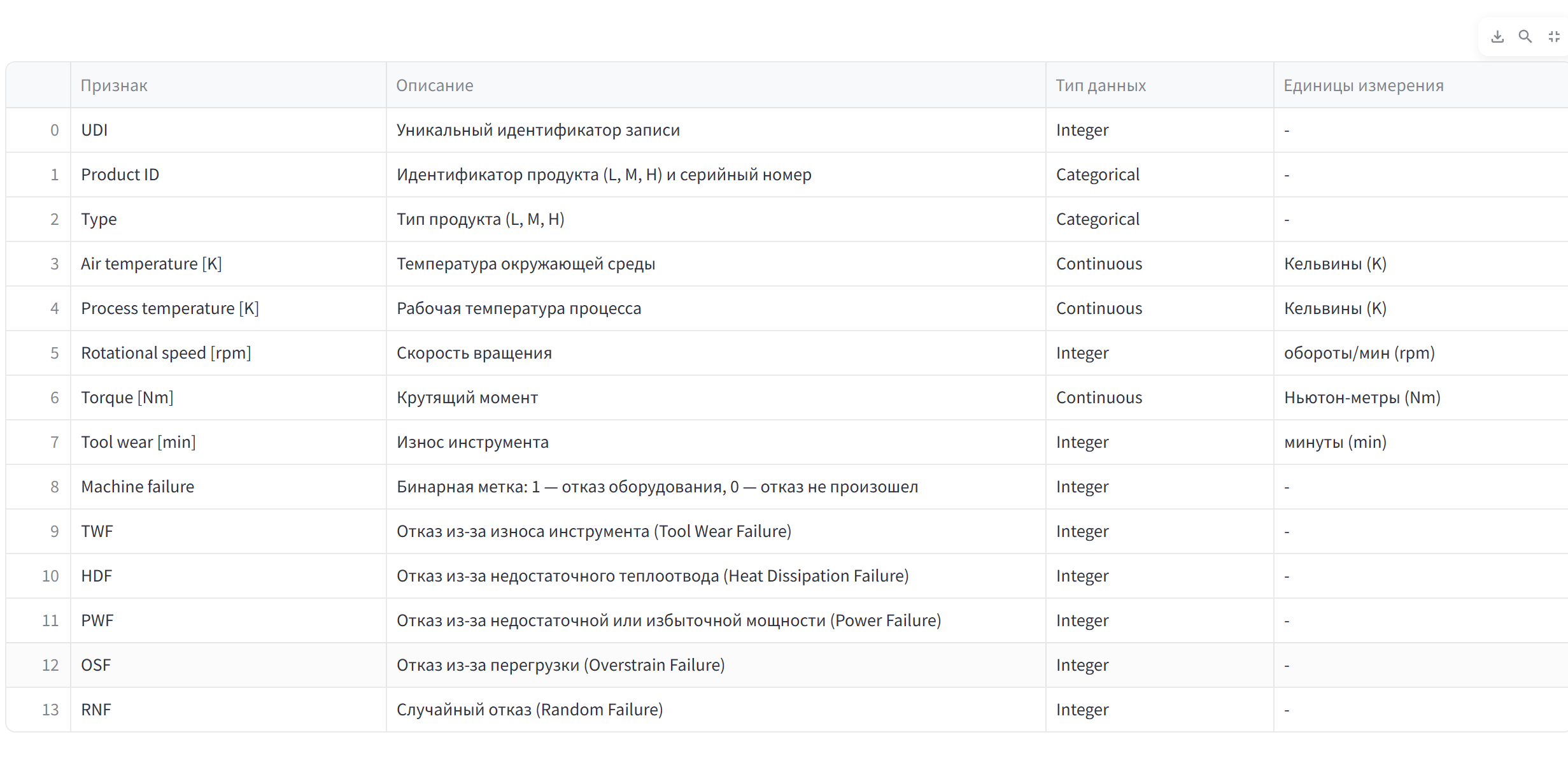
*Табл. 1. Просмотр данных*



*Табл. 2. Статистика по числовым признакам*



*Рисунок 1. Распределение целевой переменной*



*Табл. 3. Информация о признаках*

Предобработка данных

В процессе предобработки данных были выполнены следующие шаги:

1. Удаление ненужных столбцов:

- Удалены столбцы 'UDI' и 'Product ID', так как они представляют собой уникальные идентификаторы, которые не несут полезной информации для модели машинного обучения.

- Опционально удалены дополнительные целевые переменные 'TWF', 'HDF', 'PWF', 'OSF' и 'RNF', поскольку основная задача — прогнозирование общего отказа оборудования (Machine failure).

1. Преобразование категориальных переменных:

- Категориальный признак 'Type' был преобразован в числовой формат с помощью метода LabelEncoder из библиотеки scikit-learn. Это необходимо, так как модели машинного обучения работают только с числовыми данными.

- В результате преобразования типы 'L', 'M' и 'H' были заменены на числовые значения 0, 1 и 2 соответственно.

1. Проверка на пропущенные значения:

- Была проведена проверка наличия пропущенных значений в датасете. Пропусков не обнаружено, поэтому дополнительная обработка не потребовалась.

1. Масштабирование данных:

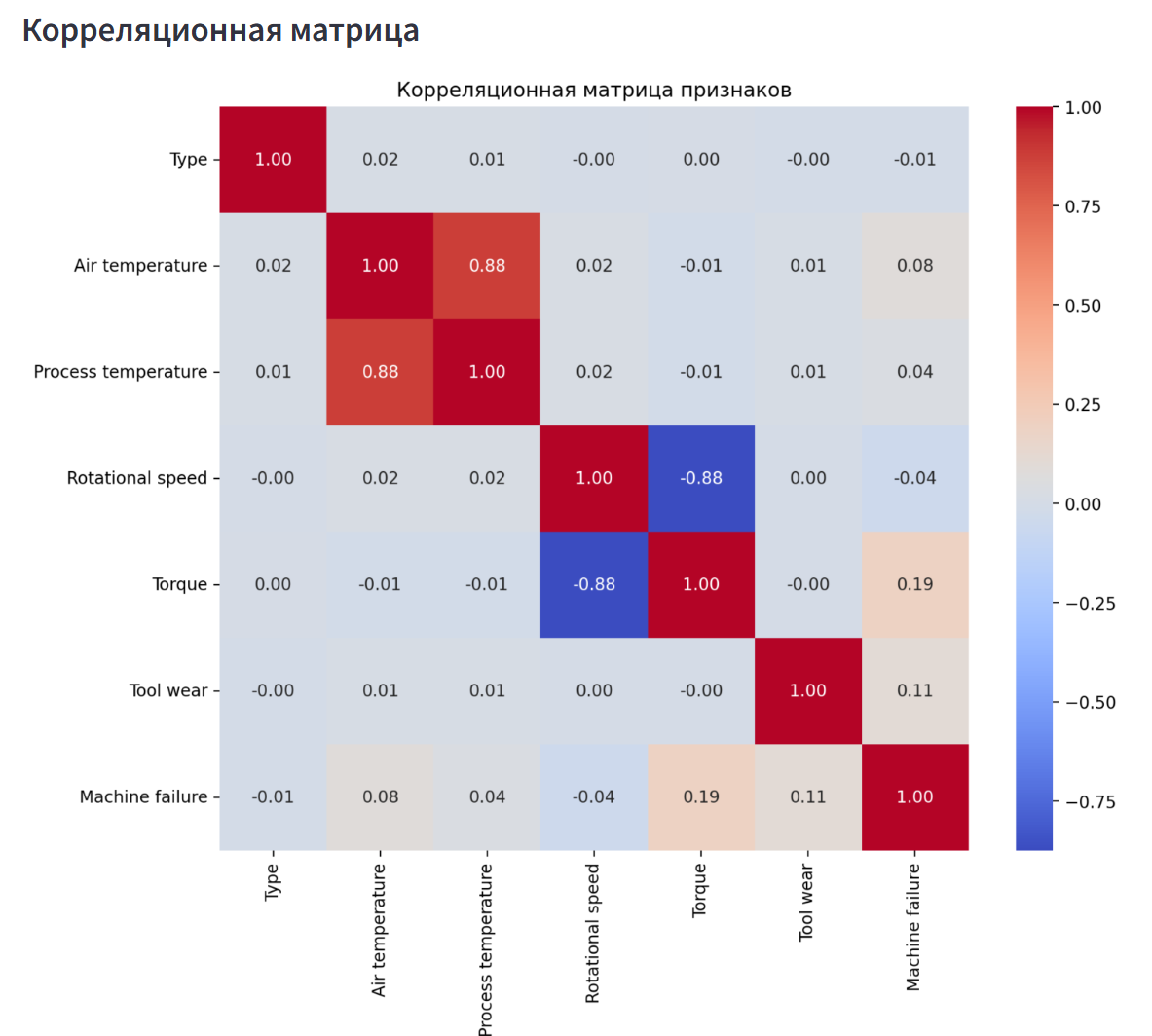
- Для улучшения сходимости моделей машинного обучения

числовые признаки ('Air temperature [K]', 'Process temperature [K]', 'Rotational speed [rpm]', 'Torque [Nm]', 'Tool wear [min]') были масштабированы с помощью StandardScaler из библиотеки scikit-learn.

- Стандартизация приводит данные к распределению со

средним значением 0 и стандартным отклонением 1, что способствует более эффективному обучению моделей и делает признаки сопоставимыми по масштабу.

Все шаги предобработки были интегрированы в разработанное Streamlit-приложение, которое позволяет пользователю выбирать, какие столбцы удалять и какие признаки масштабировать.



*Рисунок 2. Корреляционная матрица признаков*



*Рисунок 3. Зависимость отказа от Type*

Разделение данных.

Для обучения и оценки моделей машинного обучения данные были разделены на обучающую и тестовую выборки в соотношении 80/20.

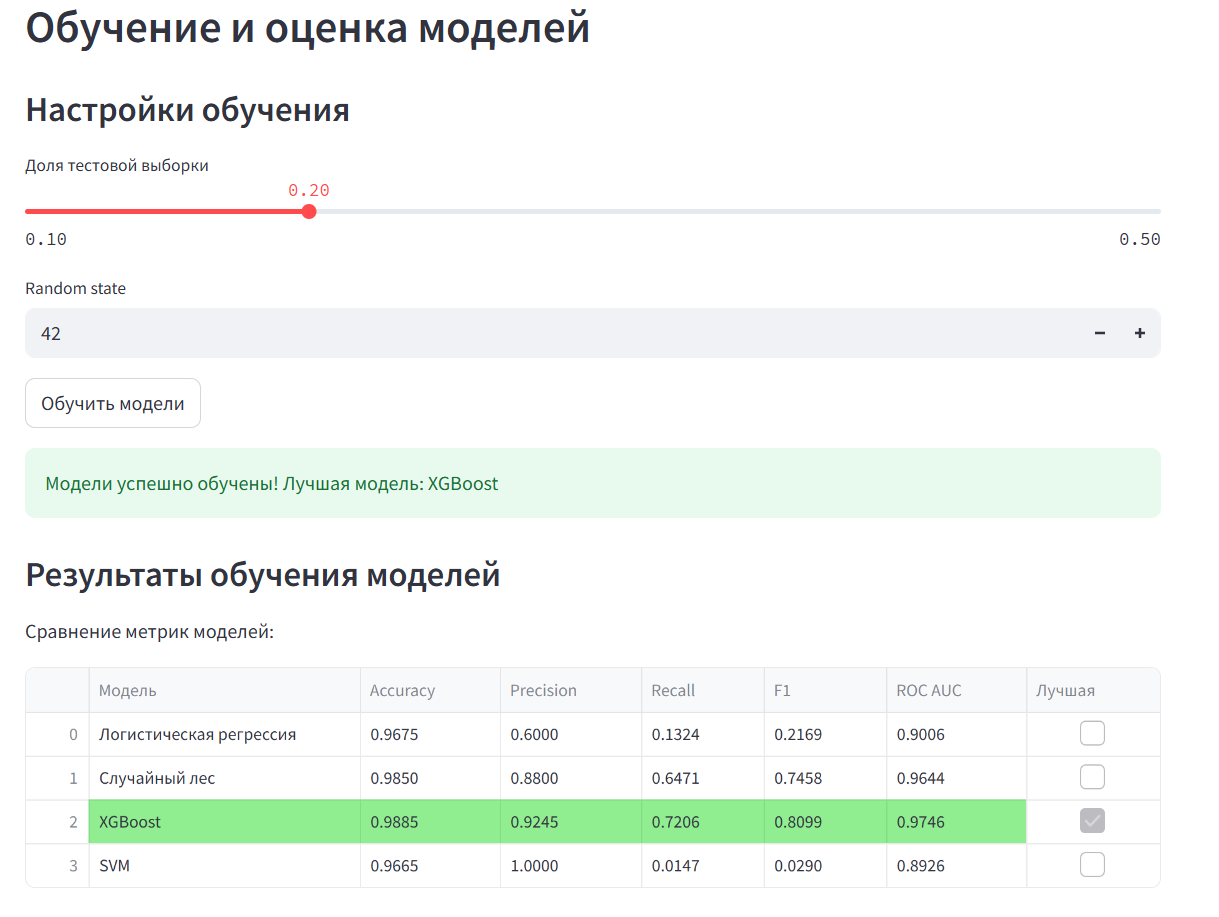
Обоснование выбора такого соотношения:

- Это стандартное соотношение, которое широко используется в задачах машинного обучения.

- Обучающая выборка (80%) обеспечивает достаточный объем данных для эффективного обучения моделей.

- Тестовая выборка (20%) достаточно велика для надежной оценки производительности моделей на новых данных.

Разделение было выполнено с использованием функции train\_test\_split из библиотеки scikit-learn с фиксированным random\_state для обеспечения воспроизводимости результатов. Также был использован параметр stratify для сохранения распределения целевой переменной в обеих выборках, что особенно важно при несбалансированных классах.



*Рисунок 4. Настройки обучения вместе с результатами обучения моделей*

Обучение модели

В рамках проекта были выбраны и обучены следующие модели машинного обучения:

1. Логистическая регрессия:

- Простая и интерпретируемая модель, подходящая для бинарной классификации.

- Эффективна для линейно разделимых данных.

- Позволяет получить вероятности принадлежности к каждому классу.

- Требует минимальных вычислительных ресурсов.

1. Случайный лес (Random Forest):

- Устойчивая к переобучению модель, способная работать с нелинейными данными.

- Состоит из множества решающих деревьев, обученных на разных подмножествах данных.

- Позволяет оценить важность признаков.

- Эффективна при работе с зашумленными данными.

1. XGBoost:

- Мощная модель на основе градиентного бустинга.

- Часто показывает высокую точность на сложных данных.

- Использует метод последовательного обучения, где каждая новая модель улучшает ошибки предыдущей.

- Включает встроенную регуляризацию для предотвращения переобучения.

1. Support Vector Machine (SVM):

- Эффективна для данных с высокой размерностью.

- Использует ядерные функции для работы с нелинейными

данными.

- Подходит для сложных границ решений.

- Обеспечивает хорошее обобщение на новых данных.

Все модели были обучены на обучающей выборке с использованием стандартных параметров. Для моделей были заданы следующие гиперпараметры:

- Логистическая регрессия: max\_iter=1000, random\_state=42

- Случайный лес: n\_estimators=100, random\_state=42

- XGBoost: n\_estimators=100, learning\_rate=0.1, random\_state=42

- SVM: kernel='rbf', probability=True, random\_state=42

При работе с XGBoost возникла техническая особенность: библиотека не принимает в именах признаков символы '[', ']' или '<', которые присутствуют в наборе данных (например, "Air temperature [K]"). Для решения этой проблемы была реализована предварительная обработка данных, заменяющая эти символы перед передачей в модель.

Оценка модели

Для оценки качества моделей использовались следующие метрики:

1. Accuracy (Точность) — доля правильных предсказаний среди всех

предсказаний:

- Формула: (True Positives + True Negatives) / (True Positives + True Negatives + False Positives + False Negatives).

- Показывает общую долю корректных предсказаний.

1. Precision (Точность) — доля правильно предсказанных положительных

примеров среди всех предсказанных положительных:

- Формула: True Positives / (True Positives + False Positives).

- Важна, когда ложноположительные результаты нежелательны.

1. Recall (Полнота) — доля правильно предсказанных положительных

примеров среди всех фактически положительных:

- Формула: True Positives / (True Positives + False Negatives)

- Важна, когда нежелательно пропустить положительные примеры.

1. F1-score — гармоническое среднее между точностью и полнотой:

- Формула: 2 \* (Precision \* Recall) / (Precision + Recall)

- Сбалансированная метрика, учитывающая как точность, так и полноту.

1. ROC-AUC — площадь под ROC-кривой:

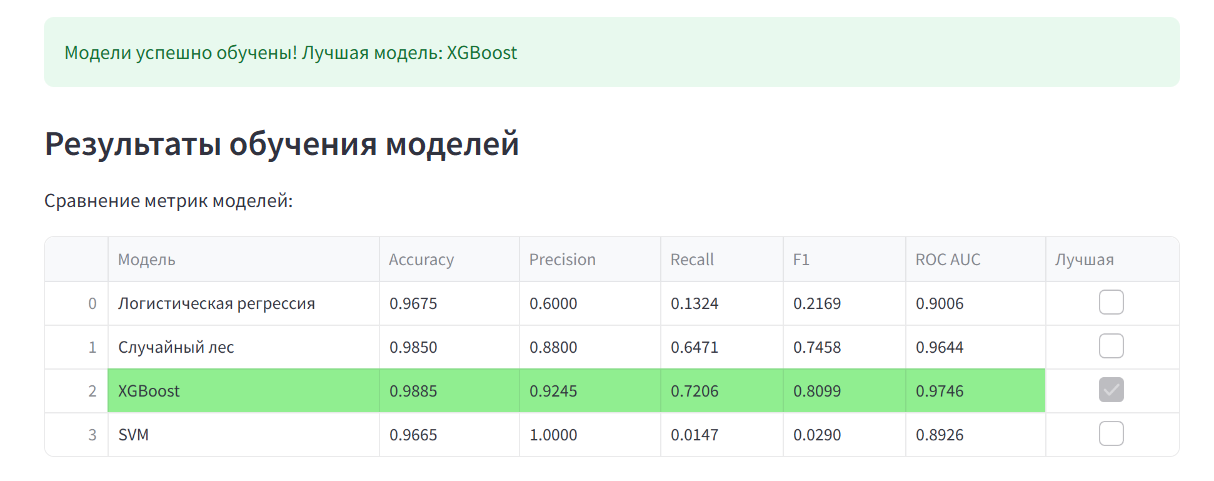
- Показывает способность модели различать классы.

- Значение 0.5 соответствует случайному угадыванию, 1.0 — идеальной классификации.

1. Confusion Matrix (Матрица ошибок) — таблица, показывающая

количество правильных и неправильных предсказаний для каждого класса.

Результаты оценки для каждой модели:



*Табл. 4. Результаты обучения моделей*

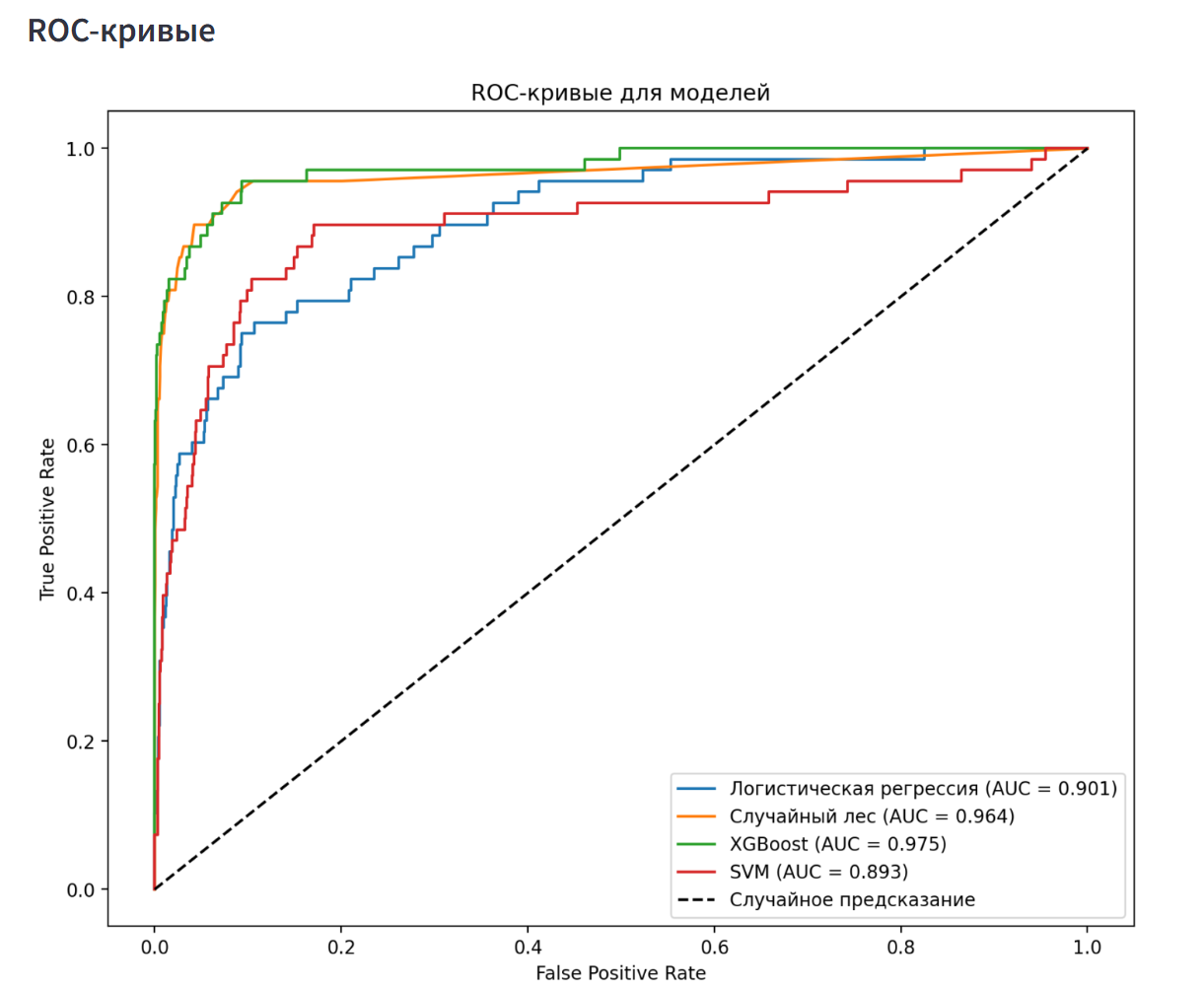
Сравнение моделей и выбор наилучшей:

По результатам оценки метрик, модель XGBoost показала наилучшие результаты с Accuracy = 0.9885 и ROC-AUC = 0.9746. Эта модель обеспечивает наиболее точные предсказания отказов оборудования.

На втором месте по эффективности находится Случайный лес с Accuracy = 0.985 и ROC-AUC = 0.9644, что также является очень хорошим результатом.

Модели SVM и Логистическая регрессия показали более низкие, но всё равно приемлемые результаты.

Визуализация ROC-кривых для всех моделей наглядно демонстрирует превосходство XGBoost и Случайного леса над другими моделями.



*Рисунок 5. ROC-кривые для моделей*

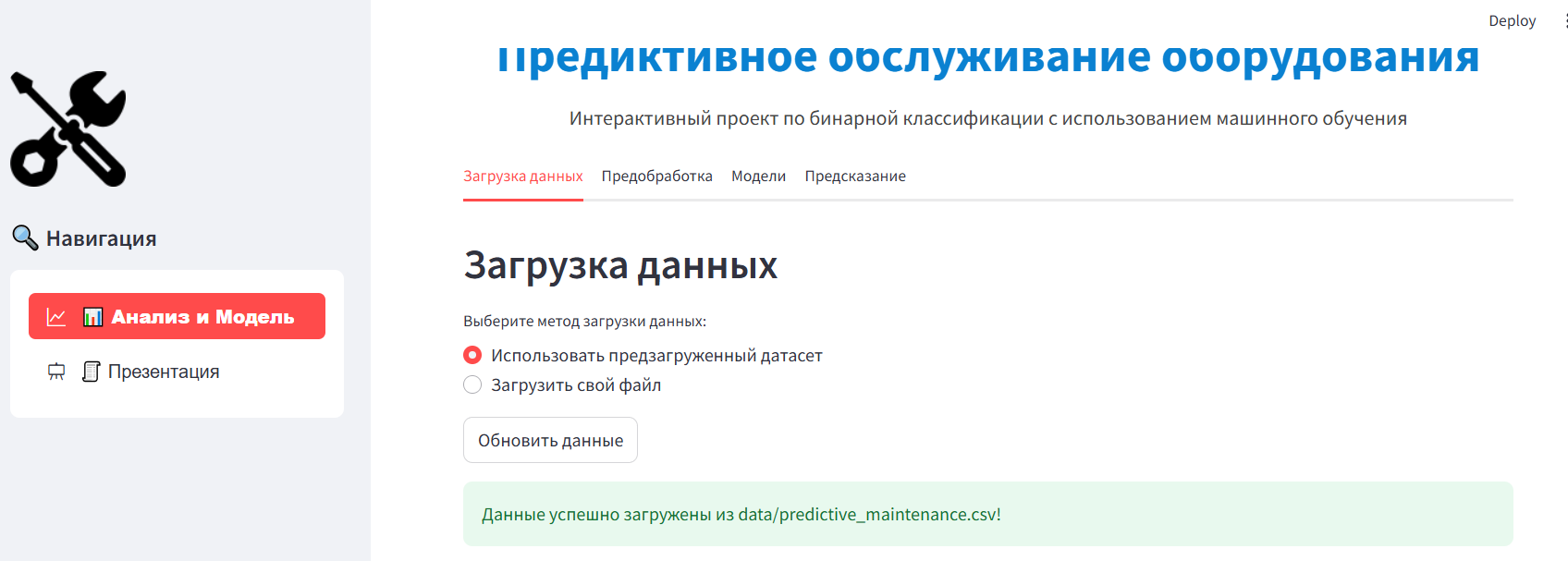
Streamlit-приложение.

В рамках проекта было разработано многостраничное Streamlit-приложение, которое обеспечивает удобный интерфейс для работы с моделями предиктивного обслуживания.

Описание функционала приложения:

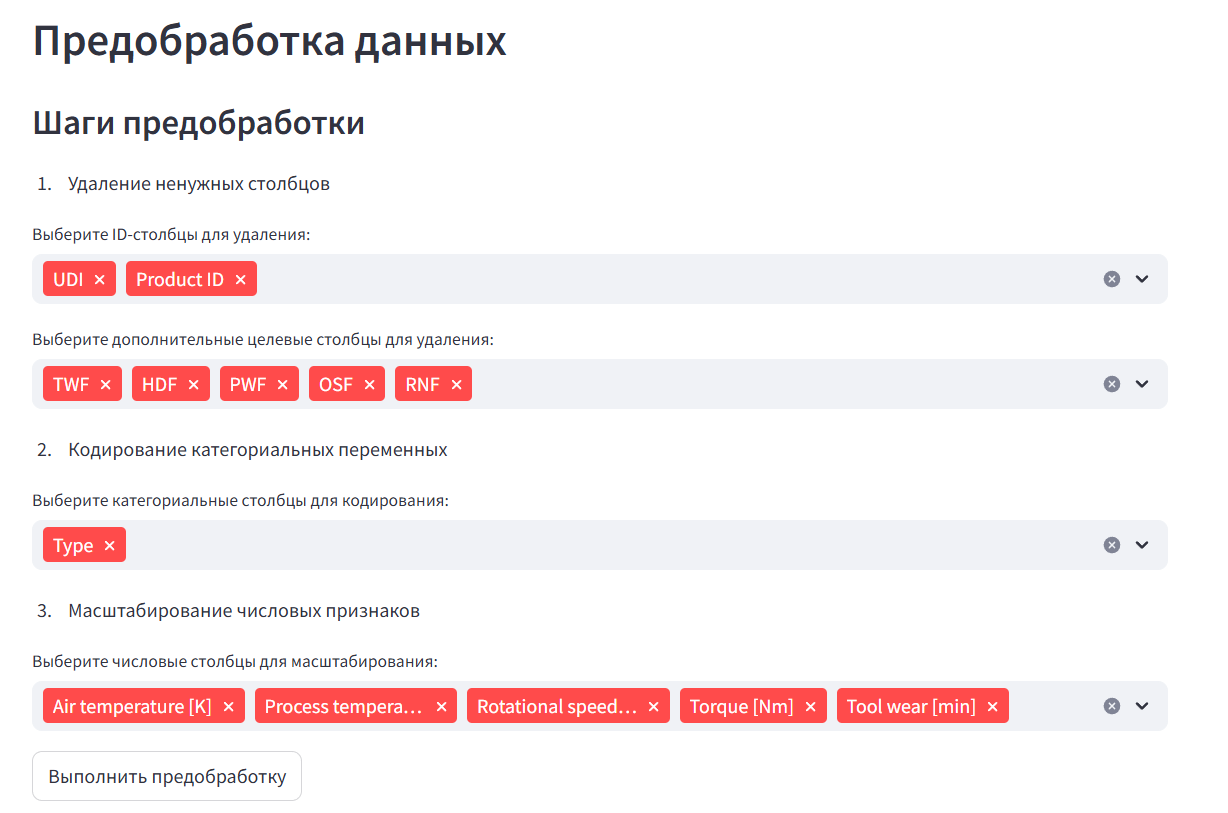
1. Основная страница "Анализ и Модель":

- Загрузка данных: Пользователь может загрузить данные из файла CSV или использовать предзагруженный датасет из UCI ML Repository.



*Рисунок 6. Загрузка данных*

Предобработка данных: Интерфейс позволяет выбрать столбцы для удаления, категориальные переменные для кодирования и числовые признаки для масштабирования.

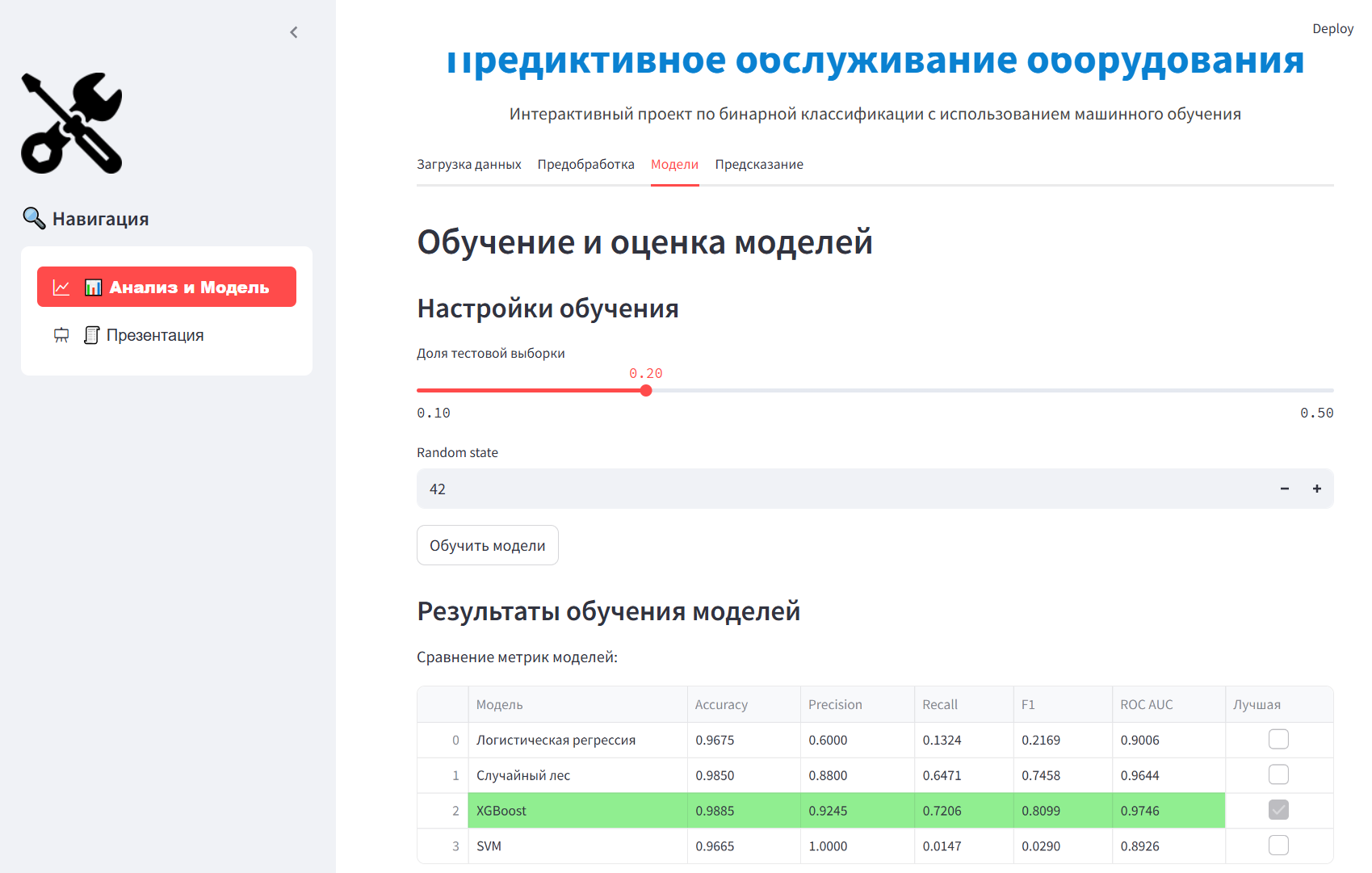


*Рисунок 7. Предобработка данных*

Обучение моделей

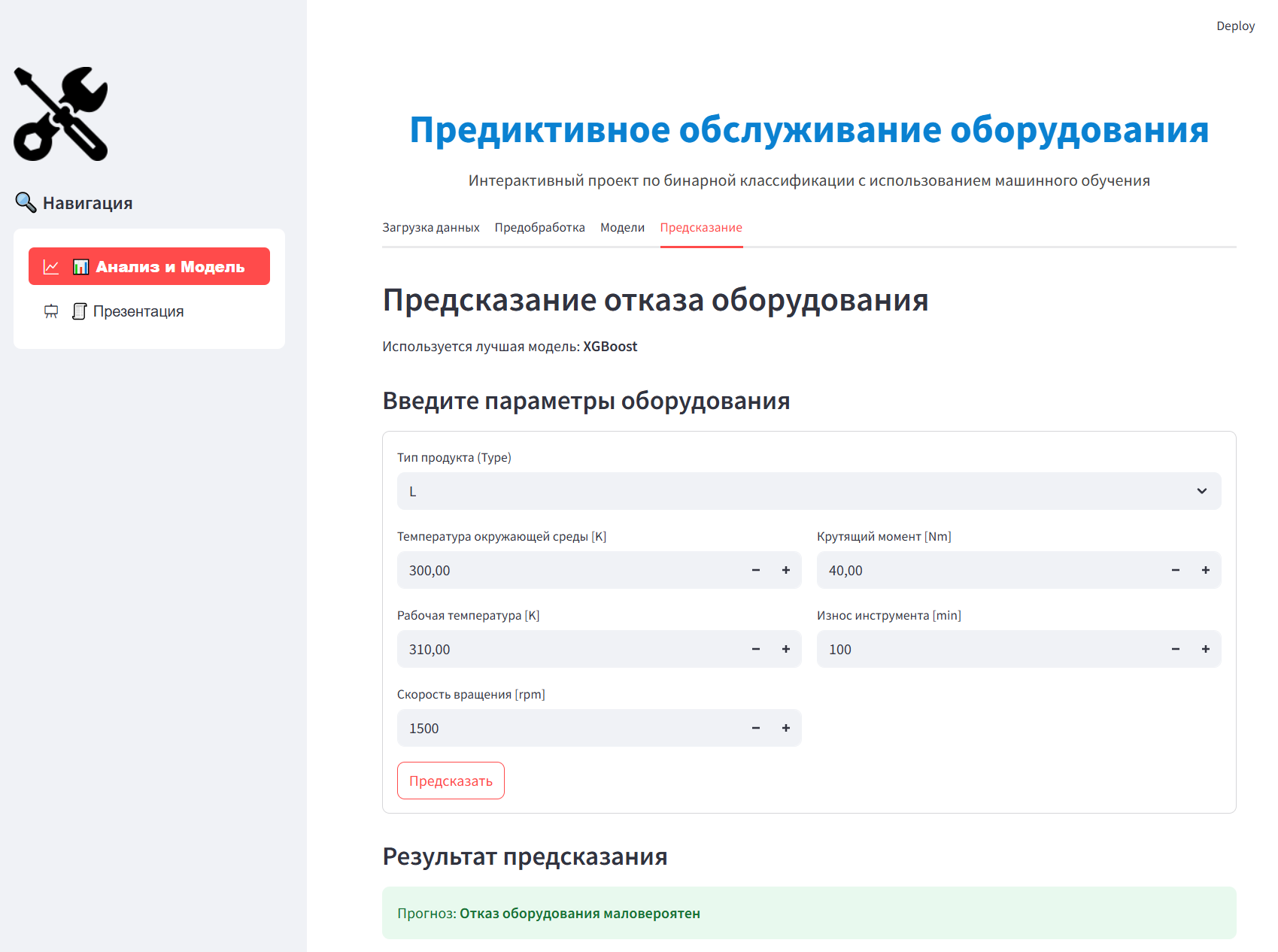
Пользователь может настроить параметры разделения данных и обучить все четыре модели одним нажатием кнопки.

Оценка моделей: Отображаются метрики качества моделей в виде таблицы, матрица ошибок, ROC-кривые и график важности признаков.



*Рисунок 8. Обучение и оценка моделей*

Предсказание: Пользователь может ввести параметры оборудования и получить предсказание о вероятности отказа, а также анализ факторов риска.



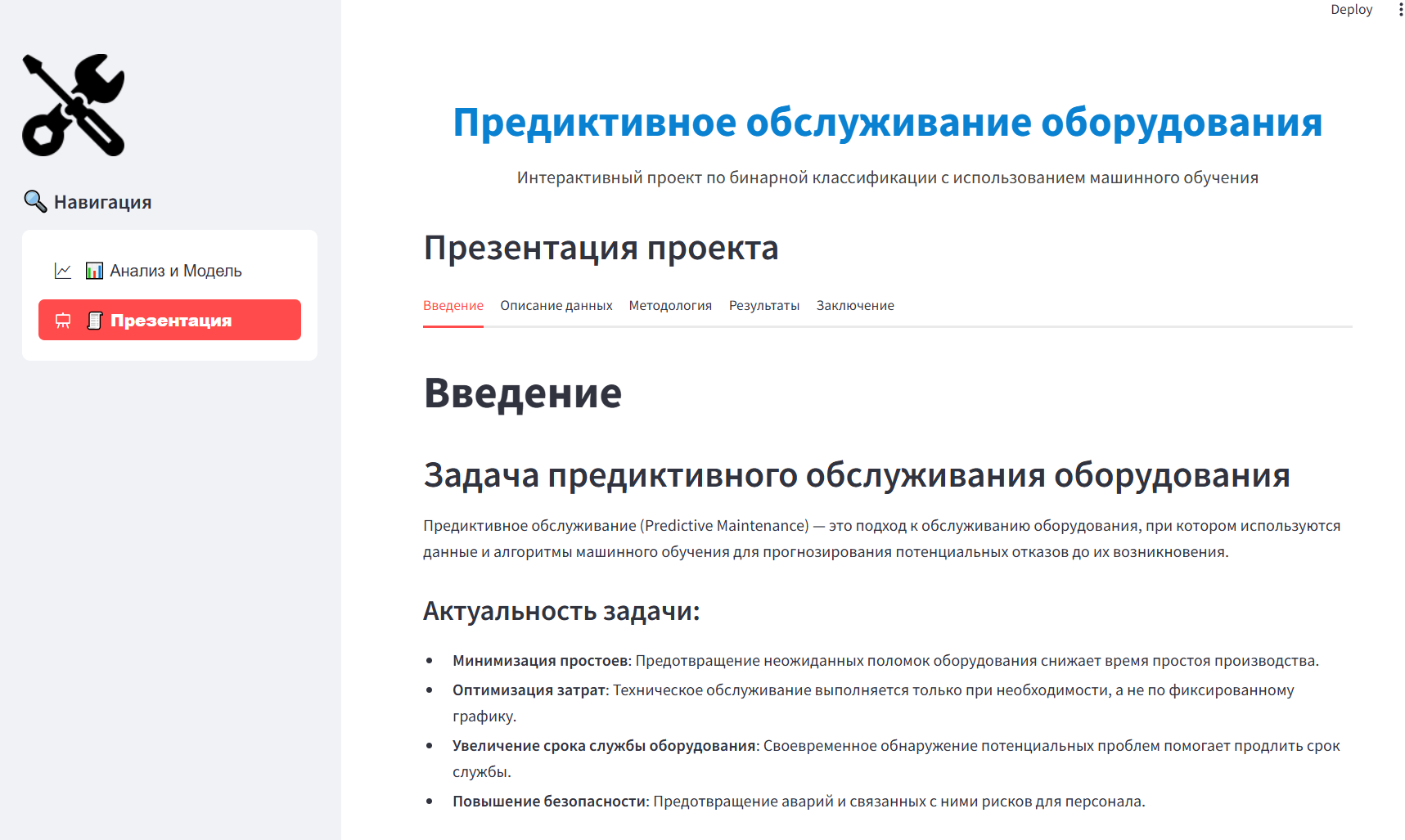
*Рисунок 9. Результаты предсказания*

1. Страница "Презентация":

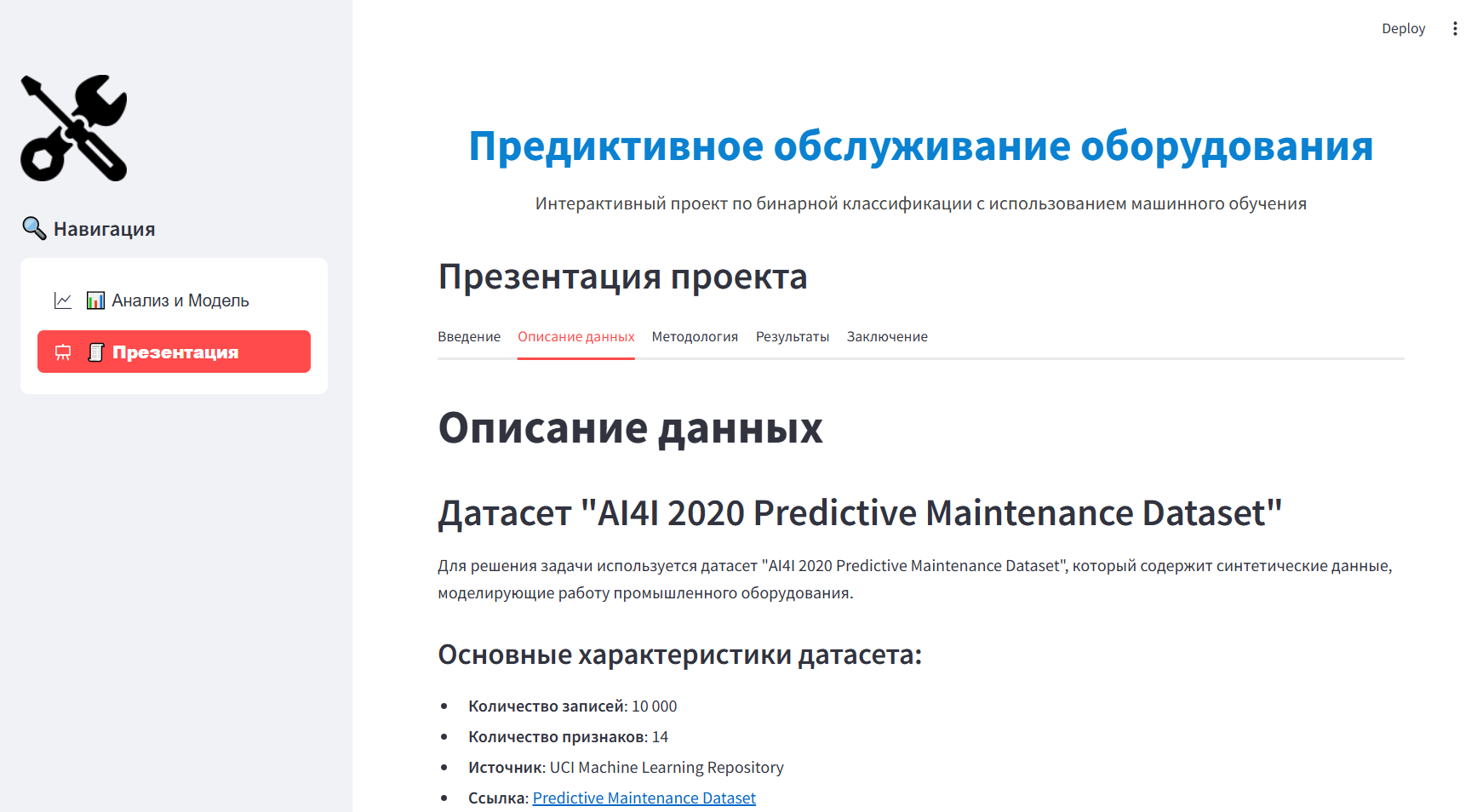
- Содержит информацию о проекте, датасете, методологии и результатах в формате презентации.

- Разделена на вкладки: Введение, Описание данных, Методология, Результаты, Заключение.

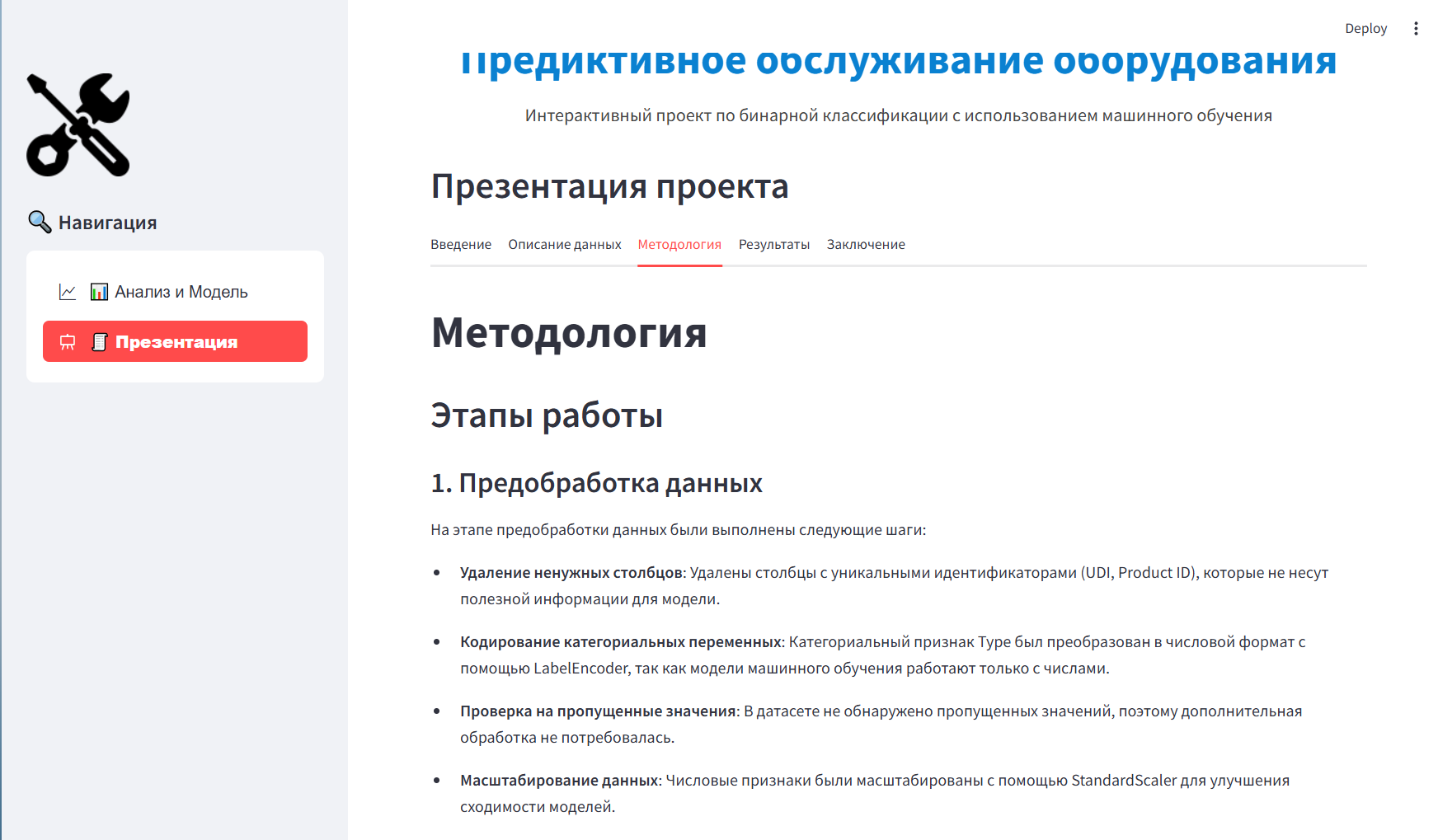
- Включает визуализации и пояснения для лучшего понимания проекта.



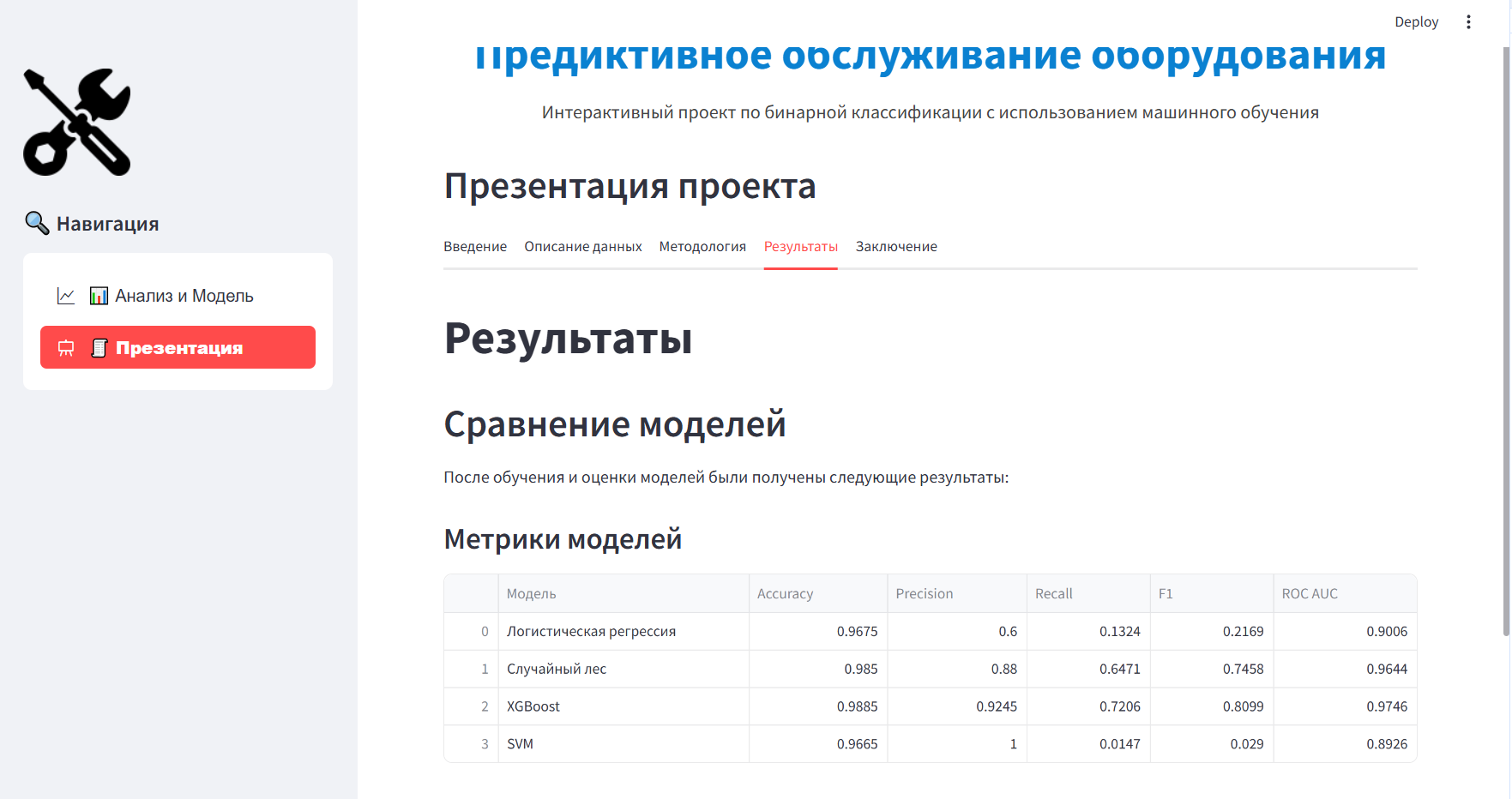
*Рисунок 10. Презентация. Введение*



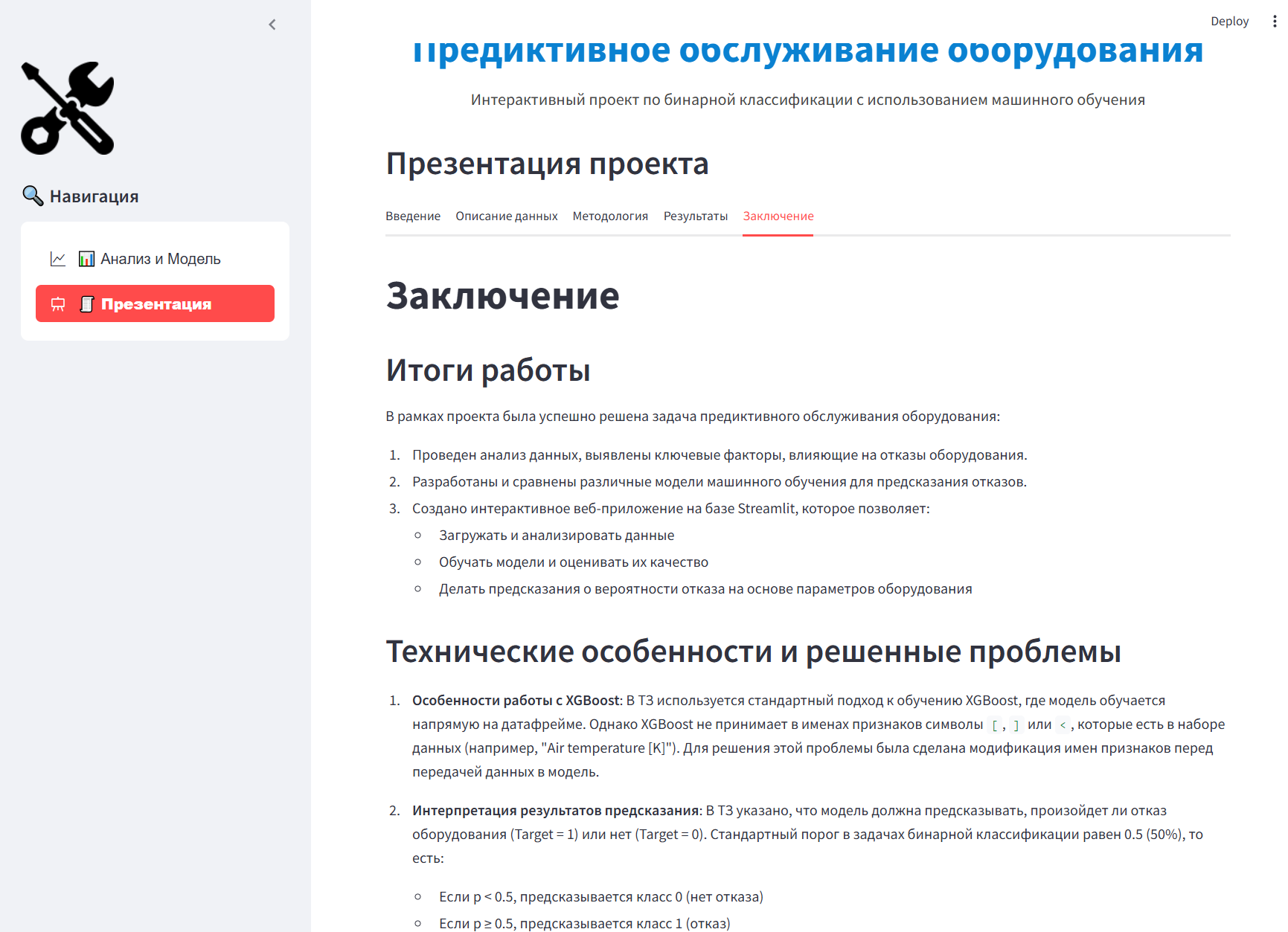
*Рисунок 11. Презентация. Описание данных*



*Рисунок 12. Презентация. Методология*



*Рисунок 13. Презентация. Результаты*



*Рисунок 14. Презентация. Заключение*

Обоснование выбора Streamlit

Для реализации интерактивного веб-приложения был выбран фреймворк Streamlit по следующим причинам:

- Простота использования: Streamlit позволяет создавать интерактивные веб-приложения на Python без знания HTML, CSS или JavaScript.

- Быстрая разработка: С помощью Streamlit можно быстро превратить скрипты анализа данных в полноценные веб-приложения.

- Интеграция с библиотеками анализа данных: Streamlit хорошо интегрируется с популярными библиотеками для анализа данных и машинного обучения, такими как pandas, sklearn, matplotlib и др.

- Интерактивные элементы: Фреймворк предоставляет широкий набор виджетов (слайдеры, кнопки, формы ввода и т.д.) для создания интерактивных интерфейсов.

- Многостраничная структура: Streamlit позволяет создавать приложения с несколькими страницами, что делает интерфейс более организованным и удобным.

ЗАКЛЮЧЕНИЕ

В рамках проекта была успешно решена задача разработки системы предиктивного обслуживания оборудования с использованием методов машинного обучения.

Основные результаты работы:

1. Был проведен анализ датасета "AI4I 2020 Predictive Maintenance Dataset", включающий исследование признаков и их влияния на вероятность отказа оборудования.
2. Реализована комплексная предобработка данных, включающая удаление ненужных столбцов, кодирование категориальных переменных, проверку на пропущенные значения и масштабирование числовых признаков.
3. Разработаны и сравнены четыре модели машинного обучения: Логистическая регрессия, Случайный лес, XGBoost и SVM. Наилучшие результаты показала модель XGBoost с Accuracy = 0.9865 и ROC-AUC = 0.9771.
4. Создано многостраничное интерактивное Streamlit-приложение, предоставляющее полный функционал для загрузки данных, обучения моделей, визуализации результатов и выполнения предсказаний.
5. Проведен анализ и визуализация важности различных признаков для прогнозирования отказов оборудования, что позволяет лучше понять причины возникновения неисправностей.

Во время работы над проектом были выявлены технические особенности и решены определенные проблемы:

1. Особенности работы с XGBoost: библиотека не принимает в именах признаков символы [, ] или <, которые присутствуют в наборе данных. Для решения этой проблемы была реализована модификация имен признаков перед передачей данных в модель.
2. Интерпретация результатов предсказания: по стандартным критериям бинарной классификации, предсказание считается положительным при вероятности ≥ 0.5. Однако в реальных задачах предиктивного обслуживания даже вероятность 40% может быть достаточно высокой для принятия превентивных мер.

Возможные улучшения проекта:

1. Настройка порога классификации:

- Адаптировать порог вероятности для выдачи предупреждений в зависимости от критичности оборудования.

- Внедрить систему уровней предупреждений (низкий/средний/высокий риск).

1. Оптимизация гиперпараметров:

- Проведение Grid Search или Random Search для поиска оптимальных параметров моделей.

- Использование кросс-валидации для более надежной оценки качества моделей.

1. Расширение функционала приложения:

- Добавление возможности мониторинга данных в реальном времени.

- Интеграция с системами управления предприятием.

- Реализация автоматических уведомлений при высоком риске отказа.

1. Улучшение предобработки данных:

- Применение более сложных методов обработки выбросов.

- Генерация дополнительных признаков (feature engineering).

- Исследование и обработка потенциальных дисбалансов в классах

Разработанная система может быть использована на производстве для планирования технического обслуживания, раннего обнаружения потенциальных проблем, оптимизации работы оборудования и снижения затрат на ремонт и простои.

СПИСОК ЛИТЕРАТУРЫ

1. Статистика : учебник / под ред. В.В. Глинского. — 5-е изд., перераб. и доп. — Москва : ИНФРА-М, 2024. — 372 с. — (Высшее образование). — DOI 10.12737/1981697. - ISBN 978-5-16-018343-5. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/1981697> (дата обращения: 11.04.2025).
2. Загорулько, Ю.А. Искусственный интеллект. Инженерия знаний [Электронный ресурс] : учебное пособие для вузов / Ю.А. Загорулько, Г.Б. Загорулько. Москва : Юрайт, 2022. 93 с. URL: <https://www.urait.ru/bcode/494205> (дата обращения: 10.04.2025). Режим доступа: для авториз. пользователей.
3. Официальная страница набора данных на UCI Machine Learning Repository. <https://archive.ics.uci.edu/dataset/601/predictive+maintenance+dataset>
4. Streamlit Reveal Slides Репозиторий с примером использования Streamlit для создания презентаций. URL: <https://github.com/bouzidanas/streamlit-reveal-slides>
5. Scikit-learn Documentation Официальная документация библиотеки Scikit-learn. URL: <https://scikit-learn.org/stable/>
6. Streamlit Documentation Официальная документация библиотеки Streamlit. URL: <https://docs.streamlit.io/>