Geekbrains

**Прогнозирование риска сердечной сердечно-сосудистых**

**заболеваний на основе методов машинного обучения и**

**искусственного интеллекта.**

Программа: Разработчик-аналитик

Специализация: Data Science в медицине

Шитко Андрей Викторович

Минск

2025

**СОДЕРЖАНИЕ**

Введение……………………………………………………………………………...2

Глава 1. Основы машинного обучения: виды и их алгоритмы………………….....5

1.1 Что такое машинное обучение…………………………………………………..5

* 1. Основные задачи машинного обучения………………………………………...6
  2. Основные виды машинного обучения…………………………………………..7
  3. Основные алгоритмы машинного обучения…………………………………...8
  4. Инструменты в машинном обучении………………………………………….14

Глава 2. Прогнозирование риска сердечной сердечно-сосудистых

заболеваний в Jupyter Notebook…………………………………………...……….16

2.1 Создание проекта и активация виртуального окружения………………….....16

2.2 Исследовательский анализ данных в Jupiter notebook………..………………16

Глава 3. Применение моделей машинного обучения для прогнозирования риска сердечной сердечно-сосудистых заболеваний (ССЗ)…………………………….35

3.1 Выбор модели машинного обучения…………………………………………..35

3.2 Модель Decision Tree…………………………………………………………...37

3.3 Модель Random Forest………………………………………………………….39

3.4 Модель Boosted Trees…………………………………………………………...44

3.5 Метод K-ближайших соседей (KNN)……………………………………….....50

3.6 Метод опорных векторов (SVM)……………………………………………….57

3.7 Модель логистической регрессии ..……………………………………………64

3.8 Сравнение методов машинного обучения…………………………………….71

3.9 Оценка рисков сердечно-сосудистых заболеваний с использованием AI (TensorFlow)………………………………………………..……………………….72

3.10 Заключительный вывод……………………………………………………….75

Список используемой литературы………………………………………………...79

Приложение…………………………………………………………………………80

**Введение**

1. Актуальность темы исследования

Сердечно-сосудистые заболевания (ССЗ) остаются ведущей причиной смертности в мире: по данным ВОЗ (2023), они ежегодно уносят 17.9 миллионов жизней, что составляет 32% всех глобальных смертей. Сердечная недостаточность (СН) — критическая стадия многих ССЗ — характеризуется прогрессирующим ухудшением функции сердца, высокой частотой госпитализаций и снижением качества жизни. В России, по оценкам Минздрава (2024), от СН страдает более 7 млн человек, а прямые экономические потери превышают 300 млрд рублей в год из-за затрат на лечение и потерь трудоспособности.

Традиционные методы диагностики СН (анализ анамнеза, ЭКГ, ЭхоКГ) эффективны, но имеют ограничения: позднее выявление (симптомы проявляются на стадии декомпенсации), субъективность интерпретации данных, высокая ресурсоемкость для массового скрининга.

В этом контексте технологии искусственного интеллекта (ИИ) и машинного обучения (МО) открывают революционные возможности для превентивной медицины. Они позволяют: выявлять скрытые паттерны в гетерогенных медицинских данных, строить индивидуальные прогнозы риска на основе мультипараметрического анализа, автоматизировать стратификацию пациентов для раннего вмешательства.

Мировой тренд на цифровизацию здравоохранения (EHR-системы, телемедицина) создает колоссальные массивы структурированных данных, пригодных для анализа МО-алгоритмами. Однако ключевая проблема — недостаток доверия врачей к «черным ящикам» ИИ — требует разработки интерпретируемых моделей, чьи решения понятны специалистам.

2. Проблема исследования

Несмотря на обилие исследований по применению МО в кардиологии, сохраняются существенные пробелы: большинство моделей валидировано на локальных выборках без проверки на внешних данных, низкая воспроизводимость результатов из-за уникальности клинических баз, пренебрежение интерпретацией прогнозов (фокус на accuracy, а не на explainability), недостаточная проработка этических аспектов (смещения данных, конфиденциальность).

3. Цель работы

Разработка и сравнительный анализ интерпретируемых моделей машинного обучения для прогнозирования риска сердечной недостаточности на основе общедоступного клинического датасета, обеспечивающих высокую точность и прозрачность принятия решений для клинического применения.

4. Задачи исследования:

А) Провести системный анализ современных методов прогнозирования СН, включая традиционные клинические шкалы (ESC-HF, CHARM) и ML-подходы.

Б) Выполнить предобработку и исследование датасета **Heart Failure Prediction Dataset (Kaggle):** очистка, анализ дисбаланса, визуализация взаимосвязей признаков.

В) Обучить и валидировать ансамбль ML-алгоритмов:

- Логистическая регрессия (базис),

- Случайный лес (Random Forest),

- Градиентный бустинг (XGBoost, LightGBM),

- Нейронная сеть (TenorFlow).

Г) Провести сравнительную оценку моделей по метрикам: Accuracy, Precision, Recall, F1-score, ROC-AUC.

Д) Реализовать интерпретацию прогнозов с помощью методов XAI:

- Анализ важности признаков (SHAP, LIME),

- Локализация вклада факторов для индивидуальных пациентов.

Е) Сформулировать рекомендации по интеграции модели в клиническую практику с учетом этических ограничений.

5. Объект и предмет исследования

Объект: Процесс прогнозирования сердечной недостаточности на основе клинико-лабораторных данных.

Предмет: Алгоритмы машинного обучения и методы их интерпретации для бинарной классификации пациентов с риском СН.

6. Научная новизна:

1. Комплексная валидация ML-моделей на открытом датасете с применением кросс-валидации и строгих метрик, ориентированных на дисбаланс классов.

2. Сравнительный анализ эффективности классических алгоритмов (Random Forest) и современных бустингов (LightGBM) для прогноза СН.

3. Глубокая интерпретация результатов с использованием SHAP-значений для выявления клинически значимых предикторов, выходящих за рамки стандартных шкал.

4. Разработка open-source прототипа с открытым кодом и документацией для воспроизведения исследования.

7. Теоретическая и практическая значимость

Теоретическая значимость:

- Уточнение роли ключевых предикторов СН (например, фракции выброса, уровня креатинина) через призму ML;

- Доказательство эффективности методов XAI для повышения доверия к ИИ в медицине.

Практическая значимость:

- Создание инструмента скрининга для терапевтов/кардиологов для раннего выявления пациентов группы риска;

- Оптимизация ресурсов: сокращение затрат на лечение поздних стадий СН на 20-30% за счет превентивных мер;

- Образовательный ресурс для медицинских вузов по применению ИИ в диагностике.

**Глава 1. Основы машинного обучения: виды и их алгоритмы**

**1.1 Что такое машинное обучение**

Машинное обучение - это раздел искусственного интеллекта, который изучает методы, алгоритмы и модели, позволяющие компьютеру обучаться на основе данных, не явно программируясь на определенные инструкции. В отличие от классического программирования, где набор инструкций задается заранее, в машинном обучении компьютер самостоятельно определяет закономерности и шаблоны в данных, чтобы делать прогнозы или принимать решения.

Используя технологию машинного обучения, программисты больше не обязаны тратить время на написание инструкций, рассматривающих все возможные сценарии и содержащих все решения. Вместо этого они могут встроить в компьютер или программу алгоритм, который самостоятельно находит решения, используя статистические данные для выявления закономерностей и предсказаний.

Технология машинного обучения на основе анализа данных впервые появилась в 1950-х годах при разработке программ для игры в шашки. За прошедшие десятилетия этот общий принцип остался неизменным, но благодаря резкому увеличению вычислительной мощности компьютеров, стало возможным создание более сложных закономерностей и прогнозов, а также решение более широкого круга задач с применением машинного обучения.

Для начала процесса машинного обучения необходимо загрузить в компьютер набор данных, на которых алгоритм будет учиться. Например, это могут быть изображения собак и кошек с уже расставленными метками, указывающими на их вид. После обучения программа сможет самостоятельно распознавать собак и кошек на новых фотографиях без меток. Чем больше данных обработала программа, тем точнее будет ее распознавание.

Сегодня благодаря машинному обучению компьютеры могут распознавать и фотографии, и изображения, и тексты. Например, программы уже умеют распознавать не только лица на фотографиях, но и пейзажи, предметы, текст и цифры. Для текстов также необходимо машинное обучение: проверка грамматики в текстовых редакторах и на мобильных устройствах уже стала стандартом. Также существуют программы, способные автоматически создавать новостные статьи на различные темы без участия человека.

**1.2 Основные задачи машинного обучения**

Все задачи, решаемые с помощью ML, относятся к одной из следующих категорий:

1) *Задача регрессии* заключается в предсказании числового значения на основе данных с различными признаками. Например, прогнозирование цены акций через определенный период времени или ожидаемый объем продаж товара на следующий месяц.

2) *Задача классификации* состоит в присвоении объекту определенной категории на основе набора признаков. Например, определение наличия определенного объекта на изображении или диагностика болезни по медицинским показателям.

3) *Задача кластеризации* предполагает разделение данных на группы по их сходству без предварительного определения категорий. Например, классификация покупателей по их покупательским привычкам или разделение текстов по тематике.

4) *Задача уменьшения* размерности заключается в сокращении количества признаков для упрощения анализа данных и улучшения визуализации. Например, сокращение размерности для отображения данных на двумерном графике.

5) *Задача выявления* аномалий заключается в обнаружении нестандартных и редких случаев в данных. Например, выявление мошеннических операций с кредитными картами или выявление необычных поведенческих шаблонов в данных.

**1.3 Основные виды машинного обучения**

Машинное обучение подразделяется на два основных типа: обучение с учителем и обучение без учителя. В обоих случаях машинам предоставляются исходные данные для анализа и выявления закономерностей, но различие заключается в участии человека в обучении.

1. *Обучение с учителем*

При обучении с учителем машине предоставляются данные вместе с ответами, что позволяет ей проверять свои гипотезы. Задача заключается в том, чтобы создать модель, которая может предсказывать целевую переменную (например, цену квартиры) на основе входных данных (площадь, расположение и т. д.). Примеры задач обучения с учителем включают регрессию (предсказание непрерывных значений, таких как цена) и классификацию (разделение объектов на категории, такие как спам или не спам).

1. *Обучение без учителя*

В обучении без учителя машине не предоставляются ответы. Она должна самостоятельно обнаруживать шаблоны и структуры в данных. Задачи обучения без учителя включают:

- Кластеризация: Разделение данных на группы на основе сходства, например, в случае распределения людей по размерным группам для пошива рубашек.

- Уменьшение размерности: Уменьшение количества признаков в данных для облегчения их отображения и анализа. Например, отображение данных с сотнями признаков в двух- или трехмерном пространстве.

**1.4 Основные алгоритмы машинного обучения**

Алгоритмы моделей машинного обучения - это математические процедуры, которые используются для обучения моделей машинного обучения на данных. Они определяют, как модель будет учиться и делать прогнозы. Существует множество различных алгоритмов машинного обучения, каждый со своими сильными и слабыми сторонами.

Вот некоторые общие категории алгоритмов машинного обучения.

1. *Дерево принятия решений для бизнеса*

Дерево принятия решений - это метод, который помогает принимать решения с учетом потенциальных последствий, эффективности и затрат. В контексте бизнес-процессов дерево принятия решений создается путем последовательности вопросов "да/нет", которые ведут к правильному выбору.

Этот метод структурирует и систематизирует проблему, обеспечивая логическую основу для принятия решений. Дерево принятия решений особенно полезно, когда необходимо учитывать несколько факторов и оценивать вероятность различных событий.

Вот как работает дерево принятия решений для бизнеса:

- Начните с определения проблемы или решения, которое нужно принять.

- Создайте корневой узел, представляющий исходную проблему.

- Добавьте ветви для каждого возможного решения или действия.

- Для каждой ветви добавьте узлы, представляющие возможные последствия, эффективность и затраты.

- Повторяйте этот процесс, создавая дочерние узлы, пока не достигнете конечных узлов, представляющих конкретные решения.

- Оцените вероятность и влияние каждого возможного пути.

- Выберите путь, который приводит к наилучшему результату с учетом всех факторов.

Дерево принятия решений помогает предприятиям принимать обоснованные решения, учитывая все релевантные факторы и возможные последствия. Оно обеспечивает прозрачный и систематический подход к принятию решений, что особенно ценно в сложных и неопределенных ситуациях.

1. *Наивная байесовская классификация*

Наивная байесовская классификация - это метод машинного обучения, используемый для классификации данных. Он основан на теореме Байеса, которая позволяет рассчитывать вероятность события на основе имеющейся информации.

Наивность в названии метода заключается в предположении, что признаки независимы друг от друга, даже если на практике это может быть и не так. Несмотря на это упрощение, наивные байесовские классификаторы часто показывают хорошие результаты в реальных задачах.

Вот как работает наивная байесовская классификация:

- Рассчитать вероятность того, что объект принадлежит каждому классу на основе имеющихся данных.

- Рассчитать вероятность того, что объект имеет конкретный набор признаков для каждого класса.

- Умножить вероятности из шагов 1 и 2 для каждого класса.

- Классифицировать объект в класс с наибольшим произведением вероятностей.

Наивные байесовские классификаторы широко используются в следующих областях:

- Фильтрация спама

- Классификация новостных статей

- Анализ настроений

- Распознавание лиц и образов

Они популярны благодаря своей простоте, эффективности и возможности работать с большим количеством признаков и данных.

1. *Метод наименьших квадратов*

Метод наименьших квадратов - это статистический метод, используемый для подгонки прямой к набору данных. Цель метода - найти прямую, которая наиболее точно соответствует точкам данных, минимизируя сумму квадратов расстояний между точками и прямой.

Как работает метод наименьших квадратов:

- Предварительная обработка данных: данные преобразуются в числовой формат, и для каждого объекта создается набор признаков.

- Определение целевой переменной: выбирается целевая переменная, которую необходимо предсказать.

- Подгонка прямой: алгоритм метода наименьших квадратов используется для подгонки прямой к данным. Алгоритм находит значения параметров прямой, которые минимизируют сумму квадратов расстояний между точками данных и прямой.

- Прогнозирование: после подгонки прямой ее можно использовать для прогнозирования целевой переменной для новых объектов.

Метод наименьших квадратов широко используется в машинном обучении для решения задач регрессии. Регрессия - это задача прогнозирования непрерывных целевых переменных (например, цены на акции или продаж).

Преимущества:

- Простой и эффективный алгоритм

- Хорошо подходит для задач регрессии с линейными зависимостями

- Может работать с большим количеством признаков и данных

Недостатки:

- Предположение о линейной зависимости может снизить точность в некоторых случаях

- Может быть чувствителен к шуму и выбросам в данных

Области применения:

- Прогнозирование спроса

- Анализ временных рядов

- Оценка рисков

- Финансовое моделирование

*4) Логистическая регрессия*

Логистическая регрессия - это метод анализа данных, который используется для предсказания вероятности возникновения определенных событий на основе одной или нескольких независимых переменных. Она применяется в различных областях, таких как кредитный скоринг, анализ рекламных кампаний, прогнозирование прибыли от продаж товаров и предсказание естественных катастроф. Логистическая регрессия позволяет оценить влияние различных факторов на вероятность наступления события и принять обоснованные решения на его основе.

1. *Метод опорных векторов(SVM)*

Метод опорных векторов (SVM) представляет собой набор алгоритмов, которые широко используются для решения задач классификации и регрессионного анализа. SVM строит гиперплоскость в N-мерном пространстве, чтобы разделить объекты на два класса. Гиперплоскость создается таким образом, чтобы максимально удалиться от ближайшей точки каждого класса.

Этот метод помогает решать разнообразные задачи машинного обучения, такие как сплайсинг ДНК, определение пола человека по фотографии, и показ рекламы на сайтах.

1. *Метод ансамблей*

Метод ансамблей в машинном обучении - это подход, при котором используется не один классификатор, а множество классификаторов, которые работают вместе для принятия решения. Этот метод снижает влияние случайных ошибок и уменьшает дисперсию результатов, так как объединение нескольких моделей дает более точные прогнозы, чем одна отдельно взятая.

Подход ансамбль учитывает различные гипотезы и позволяет расширить множество базовых гипотез для более точного прогнозирования.

1. *Кластеризация*

Кластеризация - это процесс разделения группы объектов на подмножества, называемые кластерами, таким образом, чтобы объекты в каждом кластере были более похожи друг на друга, чем на объекты из других кластеров.

Существует несколько алгоритмов для кластеризации объектов. Некоторые из них включают:

- Алгоритмы на основе центра тяжести, которые определяют кластеры на основе расстояния между объектами и их центром тяжести.

- Алгоритмы подключения, которые определяют кластеры на основе связей между объектами.

- Алгоритмы сокращения размерности, которые уменьшают размерность данных для упрощения кластеризации.

- Алгоритмы плотности, которые опираются на плотность объектов в пространстве для определения кластеров.

- Вероятностные алгоритмы, которые используют вероятностные модели для кластеризации.

- Методы машинного обучения, включая нейронные сети, для кластеризации данных.

Алгоритмы кластеризации применяются в различных областях, таких как биология (для анализа геномов и генетических данных), социология (для анализа социологических данных) и информационные технологии (для группировки данных и поиска закономерностей).

1. *Метод главных компонент*

Метод главных компонент (PCA) – это статистический метод ортогонального преобразования, который используется для преобразования наблюдений над переменными, имеющими какие-то взаимосвязи, в новый набор главных компонентов – линейно некоррелированных значений.

PCA обычно применяется для визуализации данных и сжатия информации, что позволяет упростить и уменьшить размерность данных для дальнейшего анализа. Однако PCA не подходит для данных, в которых все компоненты имеют высокую дисперсию и слабо упорядочены. Поэтому его применимость зависит от того, насколько хорошо изучена предметная область и какие данные доступны для анализа.

1. *Сингулярное разложение*

Сингулярное разложение, или SVD, в линейной алгебре представляет собой разложение прямоугольной матрицы на произведение трех матриц: U, Σ и V, где U и V являются унитарными матрицами, а Σ - диагональной матрицей.

Метод главных компонент является частным случаем сингулярного разложения. В прошлом алгоритмы компьютерного зрения основывались на этом методе, представляя лица или другие объекты как сумму базисных компонент, уменьшая их размерность и сравнивая с образцами из выборки. Современные алгоритмы сингулярного разложения в машинном обучении намного сложнее своих предшественников, но их основной принцип остается неизменным.

1. *Анализ независимых компонент*

Анализ независимых компонент, или ICA - это метод, который позволяет выявить скрытые факторы, влияющие на случайные величины или сигналы. Он строит модель, объясняющую данные с использованием независимых компонентов, которые считаются негауссовскими сигналами. В отличие от анализа главных компонент, ICA эффективнее в ситуациях, когда стандартные методы неэффективны. Этот метод широко используется в различных областях, таких как астрономия, медицина, распознавание речи, финансовый анализ и другие. Он помогает обнаружить скрытые причины явлений и является мощным инструментом для исследований и анализа данных.

**1.5 Инструменты в машинном обучении**

Инструменты машинного обучения - это программные платформы или библиотеки, которые предоставляют набор функций и алгоритмов для разработки, обучения и развертывания моделей машинного обучения. Эти инструменты позволяют разработчикам и специалистам по данным автоматизировать процессы машинного обучения и создавать более точные и эффективные модели.

Вот некоторые из наиболее популярных инструментов машинного обучения:

- TensorFlow: открытая платформа с открытым исходным кодом, разработанная Google. TensorFlow предоставляет широкий спектр инструментов и библиотек для создания, обучения и развертывания моделей глубокого обучения и других типов моделей машинного обучения.

- scikit-learn: библиотека Python с открытым исходным кодом, предоставляющая широкий спектр алгоритмов машинного обучения для классификации, регрессии, кластеризации и других задач.

- PyTorch: библиотека Python с открытым исходным кодом, разработанная Facebook. PyTorch используется в основном для глубокого обучения и предоставляет гибкую и эффективную среду для создания и обучения моделей.

- Keras: высокоуровневый API для TensorFlow, разработанный для упрощения создания и обучения моделей глубокого обучения. Keras предоставляет простой и удобный интерфейс для создания и запуска моделей с использованием TensorFlow.

- Jupyter Notebook: веб-приложение с открытым исходным кодом, которое позволяет пользователям создавать и делиться документами, содержащими живой код, уравнения, визуализации и текстовые пояснения. Jupyter Notebook широко используется для исследований в области науки о данных и разработки моделей машинного обучения.

Другие популярные инструменты машинного обучения:

- XGBoost: библиотека с открытым исходным кодом для градиентного бустинга, используемая для классификации и регрессии.

- LightGBM: быстрая и эффективная библиотека градиентного бустинга, используемая для классификации и регрессии.

- CatBoost: библиотека с открытым исходным кодом для градиентного бустинга, оптимизированная для категориальных данных.

- H2O.ai: коммерческая платформа машинного обучения, предоставляющая широкий спектр алгоритмов и инструментов для создания и развертывания моделей машинного обучения.

- Azure Machine Learning: облачная платформа машинного обучения от Microsoft, предоставляющая управляемую среду для разработки, обучения и развертывания моделей машинного обучения.

Эти инструменты машинного обучения предоставляют широкий спектр функциональных возможностей, включая:

- Предварительная обработка и очистка данных

- Выбор признаков и уменьшение размерности

- Обучение и оценка моделей машинного обучения

- Оптимизация гиперпараметров

- Развертывание и мониторинг моделей

- Визуализация и отчетность

Инструменты машинного обучения значительно упрощают и ускоряют процесс разработки и развертывания моделей машинного обучения, позволяя специалистам по данным и разработчикам создавать более точные и эффективные модели для широкого спектра приложений.

**Глава 2. Прогнозирование риска сердечной сердечно-сосудистых заболеваний в Jupyter Notebook.**

**2.1 Создание проекта и активация виртуального окружения**

Теперь мы приступили к основной нашей практической работе. Вся наша работа будет вестись в Jupiter notebook с автоматическим использованием языка программирования Python.

Создадим на нашем компьютере директорию “Итоговая работа 2 год”. Поместим туда файл heart.csv, скачанный из сайта [www.kaggle.ru](http://www.kaggle.ru). В данном файле содержится информация об медицинских показателях исследуемых пациентов, которые имеют предрасположенность с сердечно-сосудистым заболеваниям (ССЗ).

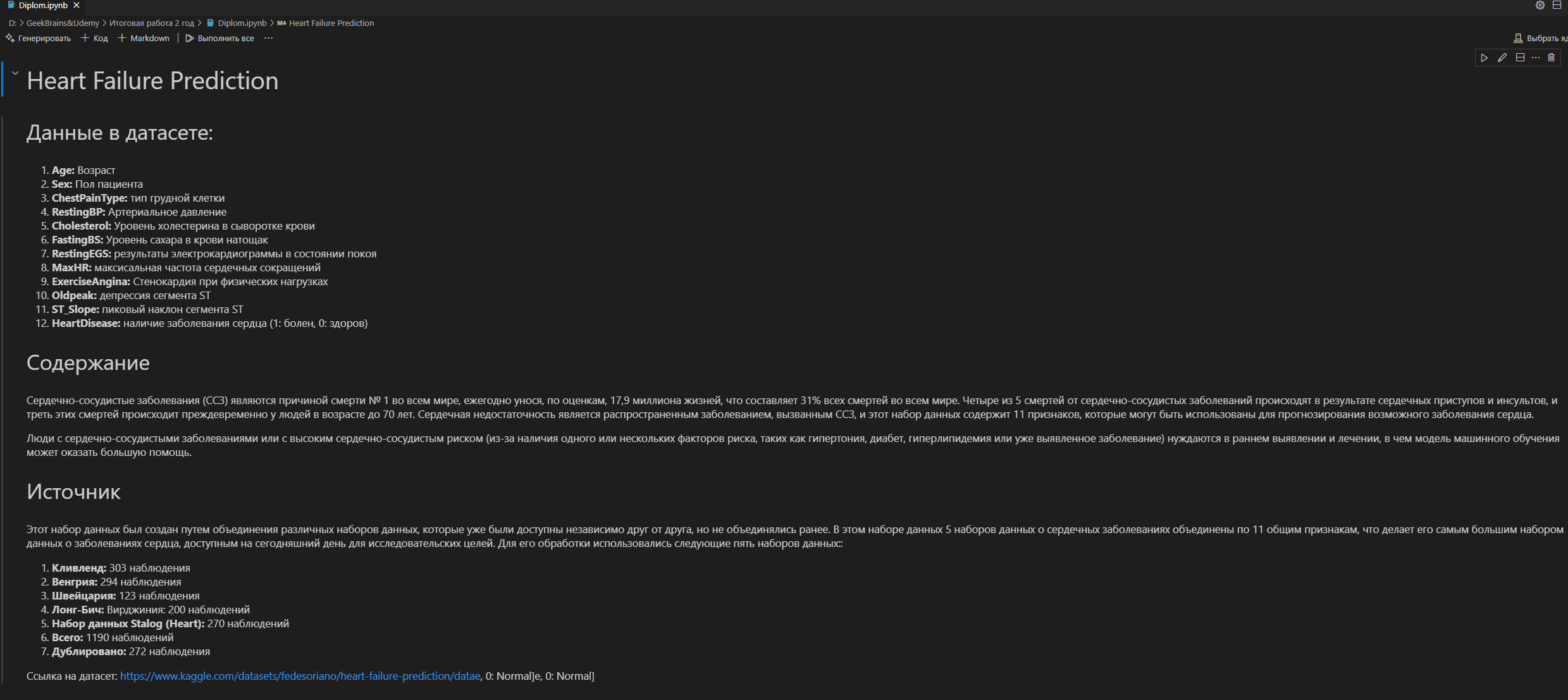
Запустим Jupiter notebook, перейдём в нашу вновь созданную директорию и создадим там файл Diplom.ipynb, где и будет выполнятся анализ данных. Все файлы будут лежать в одной директории. По умолчанию будет использоваться виртуальное окружение Conda, являющееся изолированной средой Python, в которой установлены определенные версии.

**2.2 Исследовательский анализ данных в Jupiter notebook.**

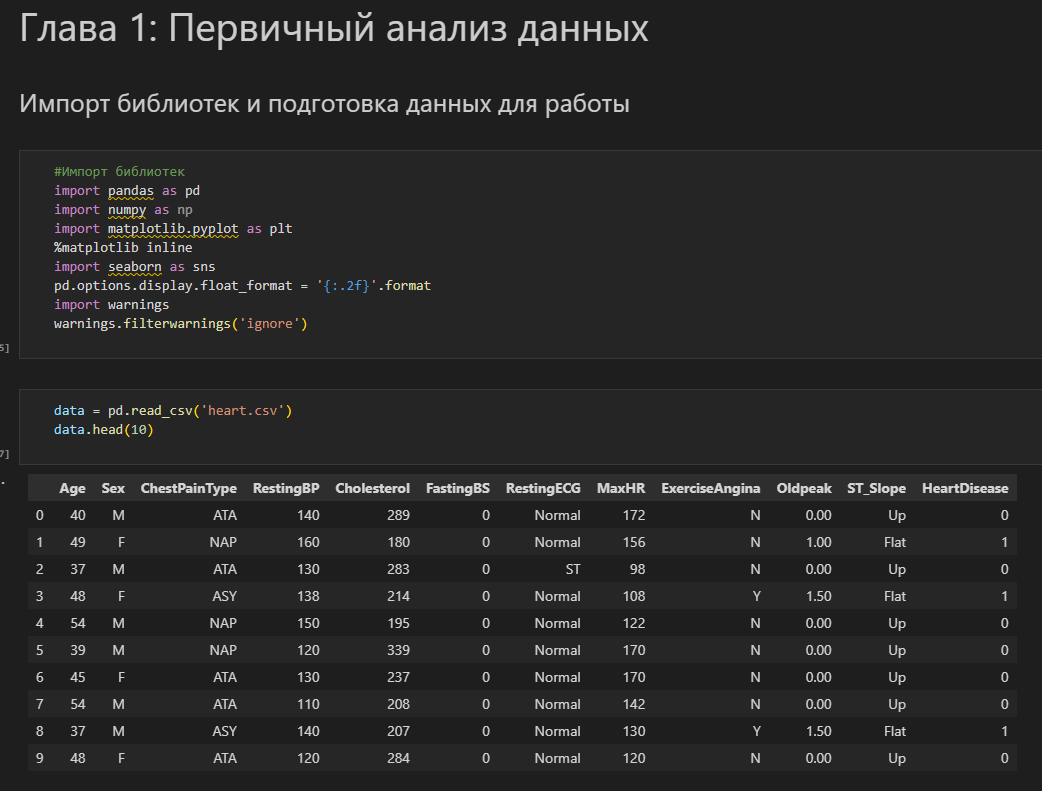
Мы приступаем к исследовательской части нашей работы. Откроем созданный файл Diplom.ipynb, и всю работу будем ввести в этом файле. У нас будет очень много кода на языке программирования Python. Поэтому в целях удобства и читаемости мы будем вставлять последовательно фото (скриншот) части нашего кода, выполненного в Jupiter notebook и описывать его. Однако скриншоты для удобства я буду через VS Code, потому что для этого не требуется запускать Anaconda Navigator.

Начнём наше исследование. Ещё раз напомним, что целью нашей работы является создание модели предсказания того, какие пациенты имеют наиболее высокий риск заболеть ССЗ на основании лабораторных и функциональных, а также их возраста. К этому мы будем двигаться постепенно, анализируя различные показатели и метрики.

Опишем обозначения из файла Diplom.csv.



Импортируем данные, загрузим датасет:

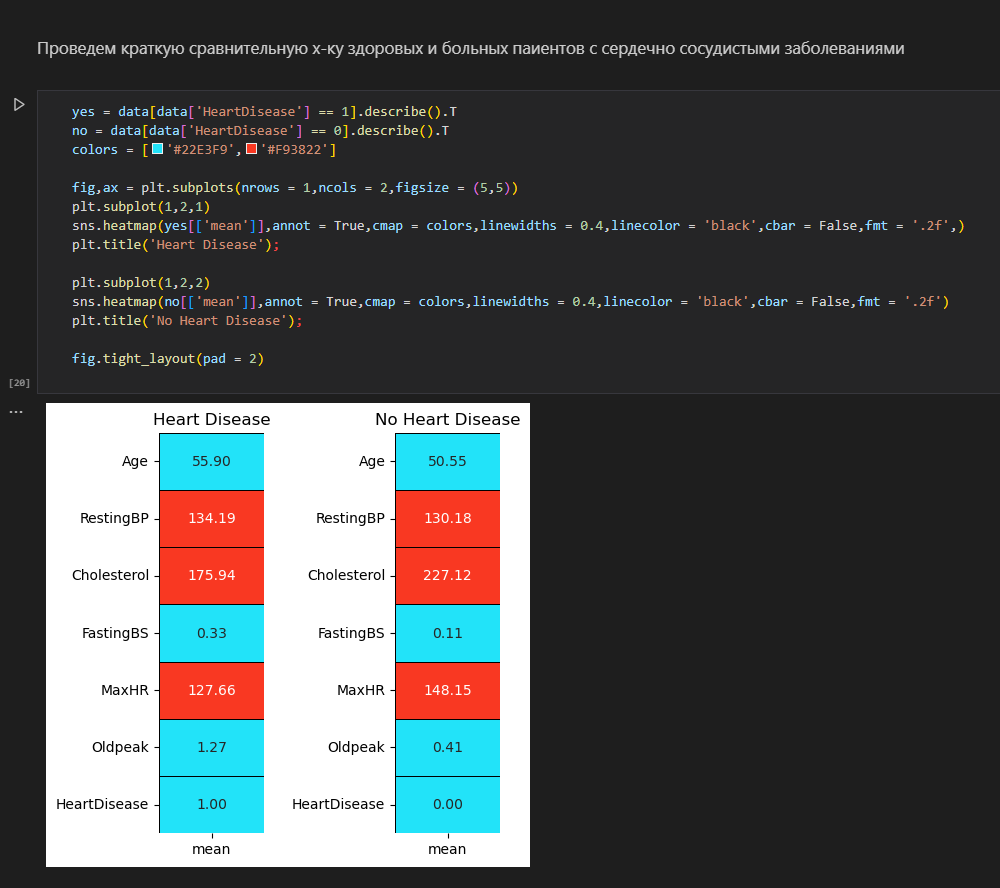


Проведем общий обзор представленных данных:



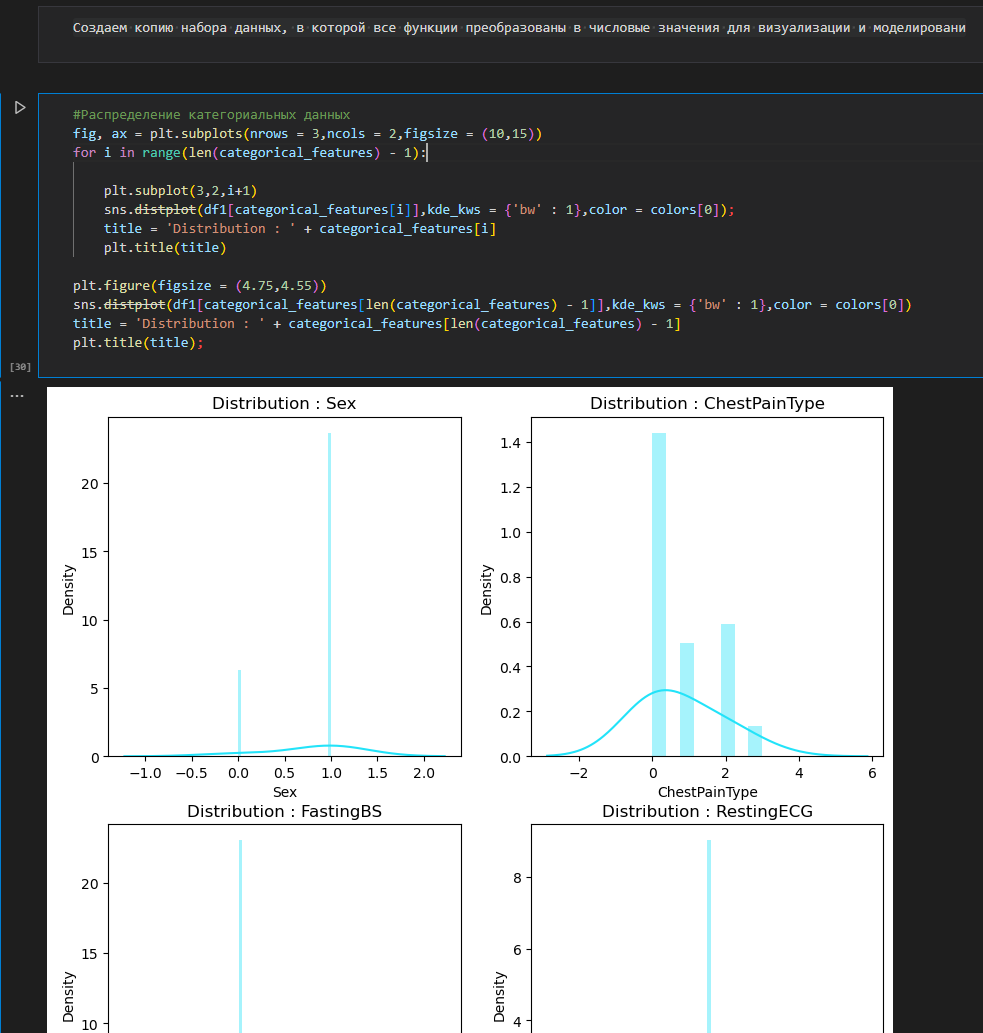
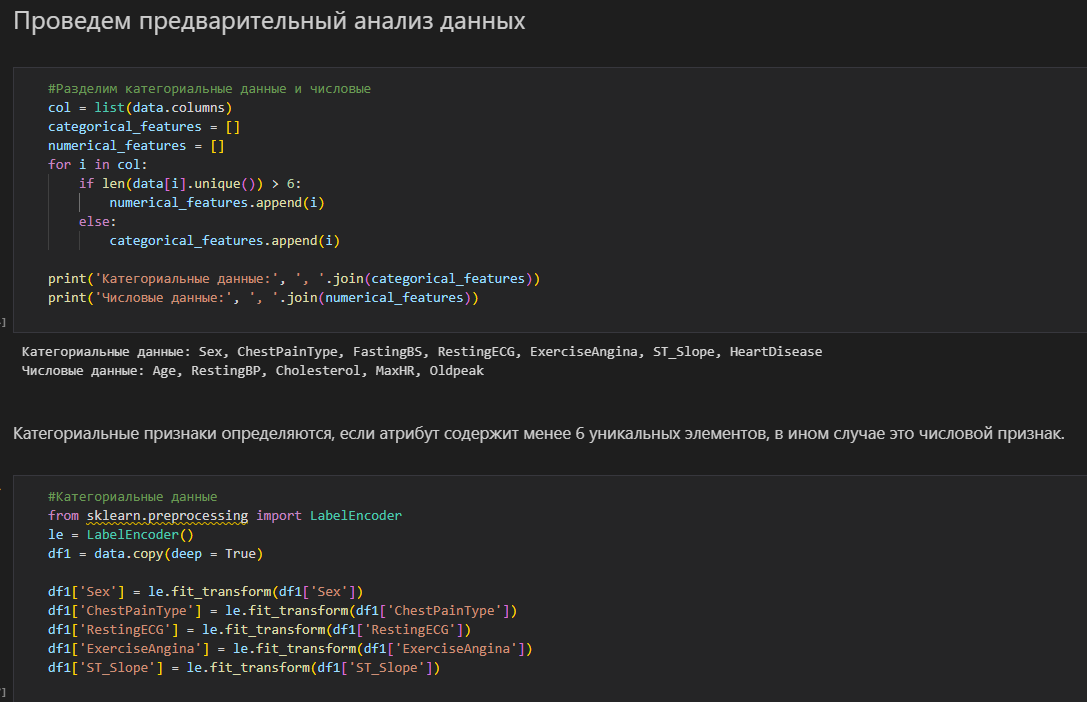
Через запрос *info()* видно, что в датасете отсутствуют пропуски во всех строках данных. Далее в самом коде есть подсчет null значений с помощью команды *data.isnull().sum()*, которая подтверждает наше наблюдение.

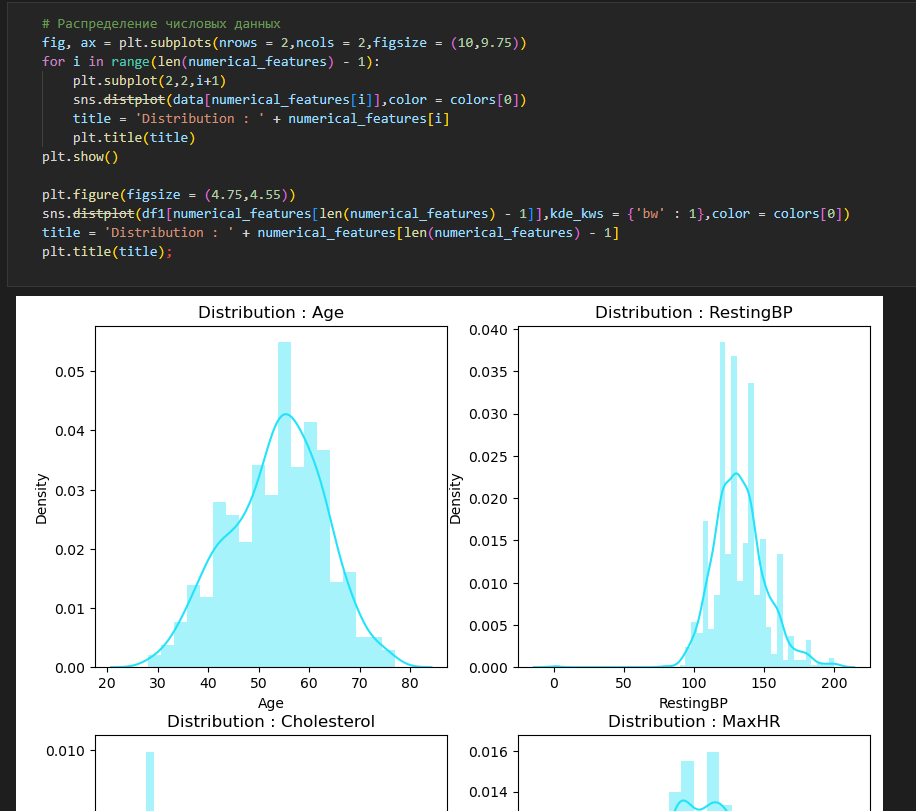
Далее проведем сравнительную характеристику данных между здоровыми пациентами и теми, кто болен ССЗ. Мы наглядно сравним возраст, артериальное давление, уровни холестерина, уровень сахара, частоту сердечных сокращений при нагрузке, депрессию сегмента ST.



Без анализа мы видим значимое отличие в показателях холестерина в крови, а также в депрессии сегмента ST (Oldpeak).

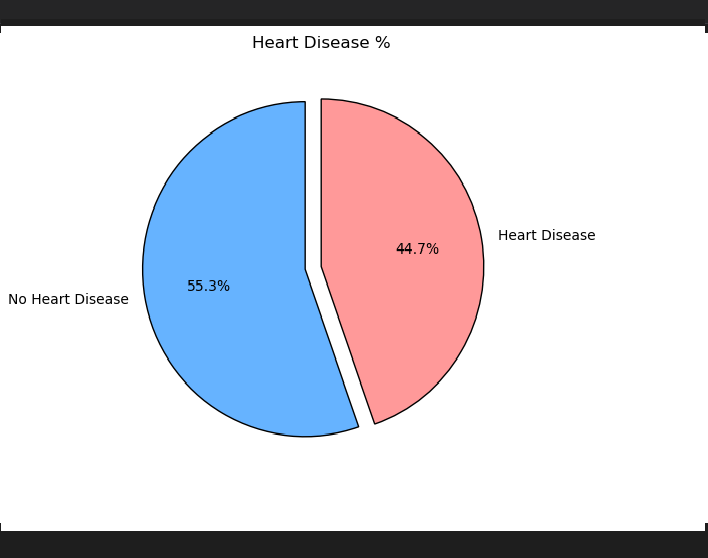
Для дальнейшего анализа мы разделим данные на категориальные и числовые. После разделения данных мы видим, что все категориальные признаки распределены нормально. Числовые данные тоже распределены нормально кроме *Oldpeak* (асимметричное распределение с длинным хвостом вправо) и *Cholestrol* (бимодальное распределение, то есть два пика частоты.):

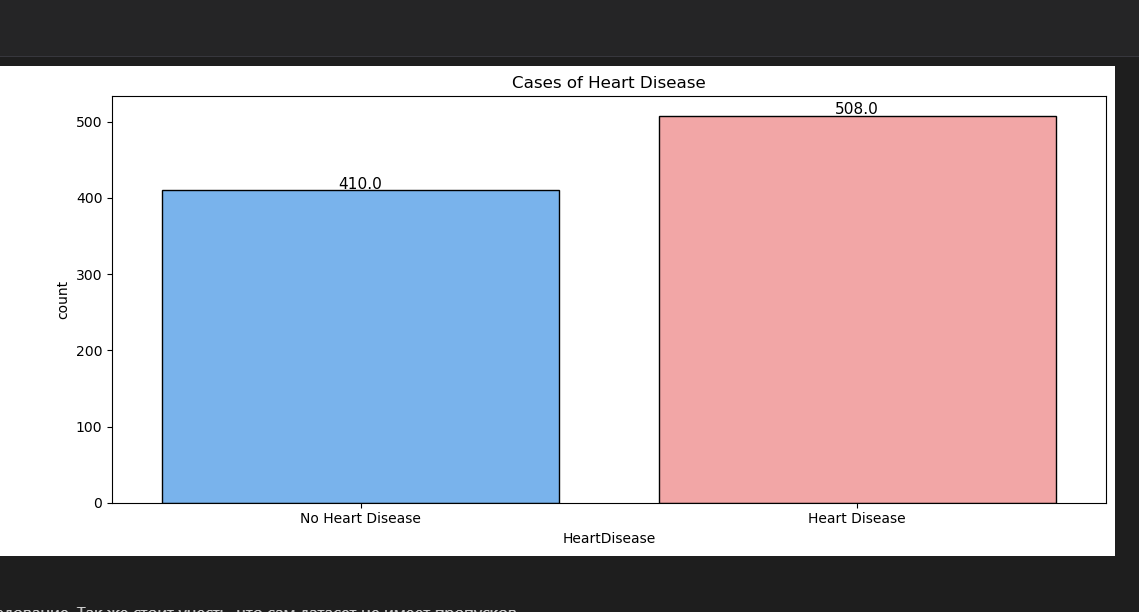




Все графики представлены в Diplom.csv.

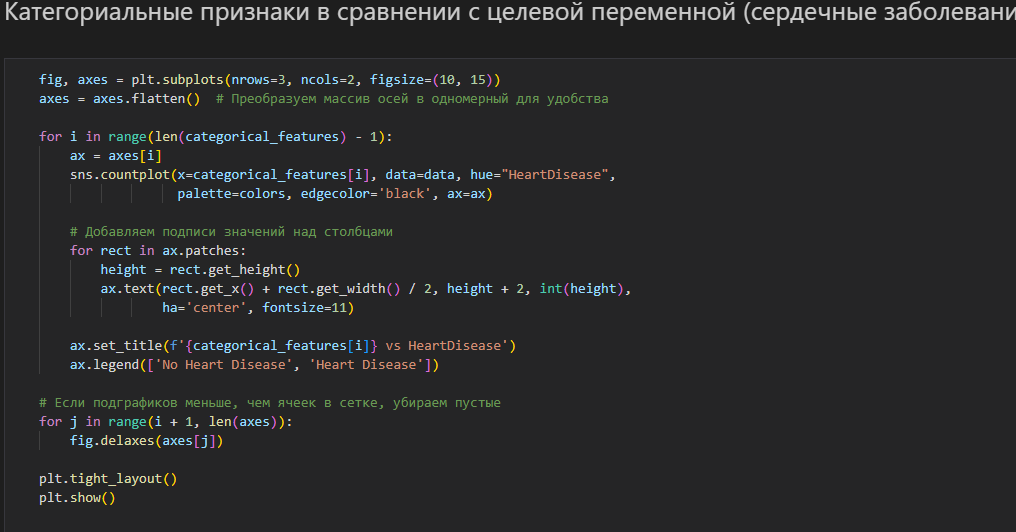
Далее мы визуализируем целевую переменную *Heart Disease* и посмотрим равномерно ли распределены данные. В скринщоте ниже представлены круговая и столбчатые диаграммы:

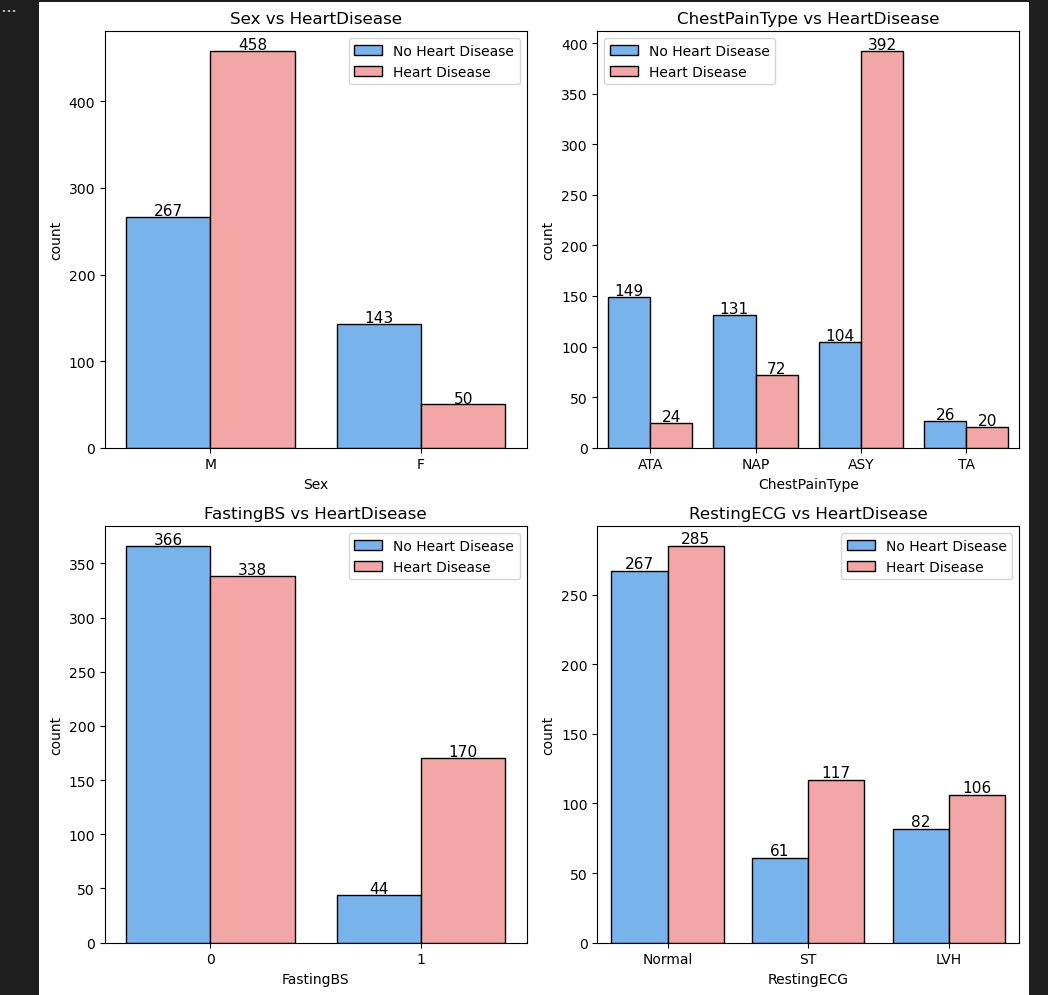


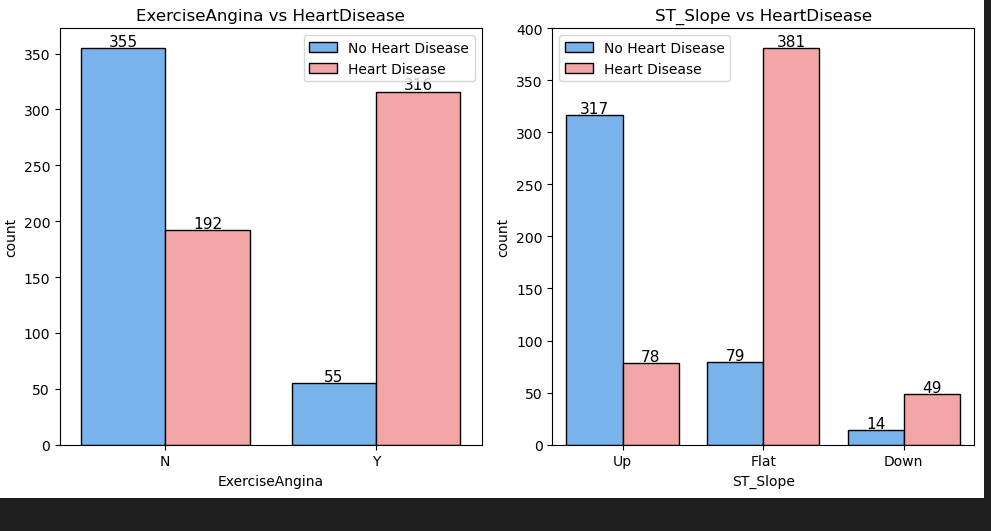


Как видим данные распределены относительно равномерно, то позволит провести более точное исследование. Так же стоит учесть, что сам датасет не имеет пропусков.

Дальше мы будем сравним все категориальные признаки с нашей целевой переменной, ниже будет представлен код, а затем графики.







**Мы можем сделать выводы:**

1) Мужское население имеет больше пациентов с сердечными заболеваниями, чем без них. В случае женского населения пациентов с сердечными заболеваниями меньше, чем без них.

2) Тип боли в груди ASY явно указывает на высокую вероятность сердечного заболевания.

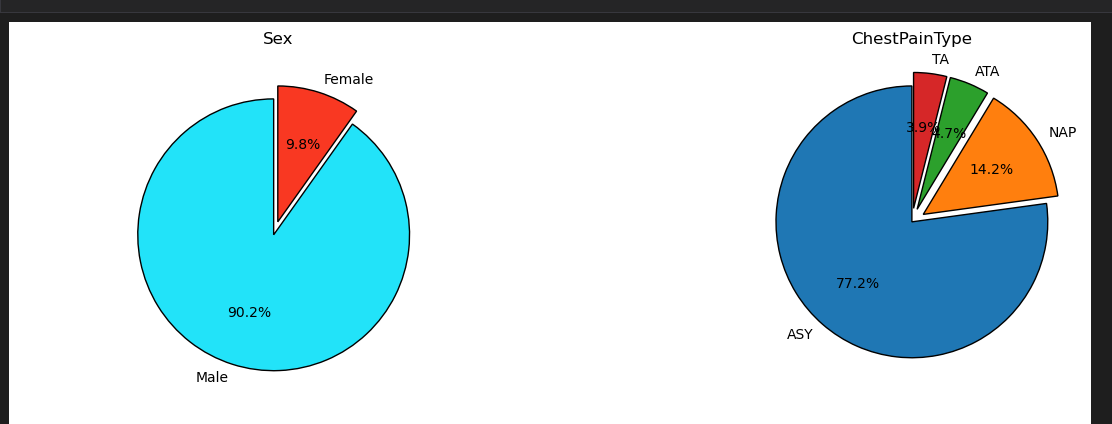
3) Показатель уровня глюкозы натощак (Fasting Blood Sugar) является спорным признаком! Пациенты как с повышенным, так и с нормальным уровнем глюкозы натощак имеют значительное количество случаев сердечных заболеваний.

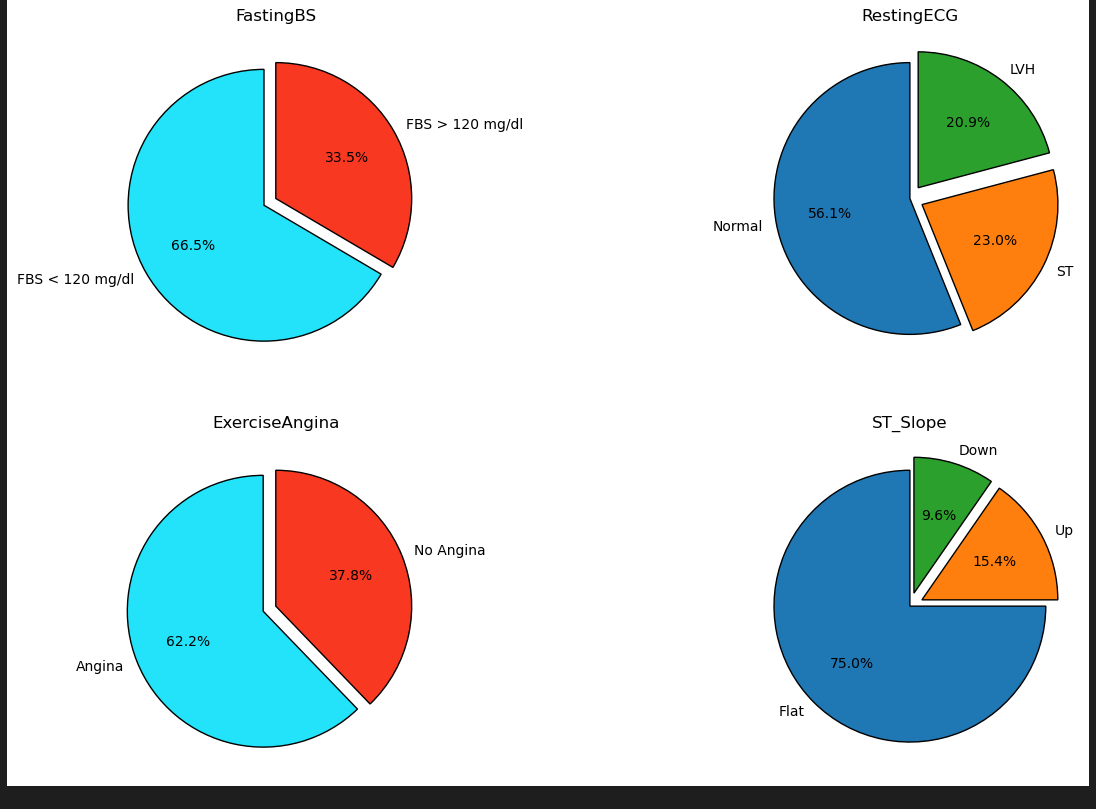
4) Результаты электрокардиограммы в состоянии покоя (RestingECG) не демонстрируют чёткой категории, выделяющей пациентов с сердечными заболеваниями. Все три варианта имеют большое количество таких пациентов.

5) Физическая нагрузка, вызывающая стенокардию (Exercise Induced Angina), однозначно повышает вероятность диагноза сердечного заболевания.

6) Что касается показателя ST\_Slope, то flat slope демонстрирует очень высокую вероятность диагноза сердечного заболевания. Наклон вниз down также показывает такой же результат, но на очень небольшом числе данных.

Далее мы определим какие категориальные признаки наиболее сильно влияют на риски ССЗ:





**Сделаем выводы:**

1. Из всех пациентов с сердечными заболеваниями значительно преобладают мужчины ~ 90%.

2. Что касается типа боли в груди, то тип ASY преобладает — на него приходится 77% случаев, ведущих к сердечным заболеваниям.

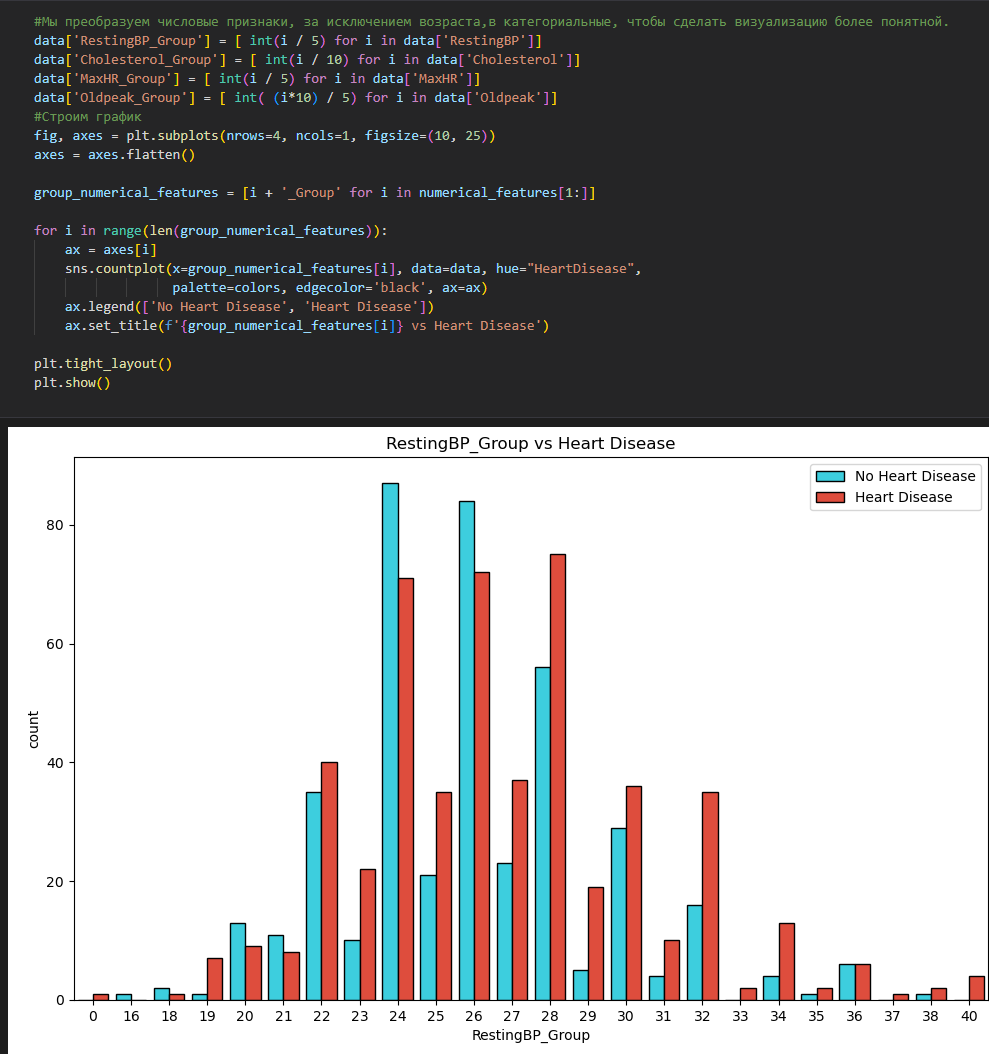
3. Уровень глюкозы натощак ниже 120 мг/дл показывает высокую вероятность развития сердечных заболеваний.

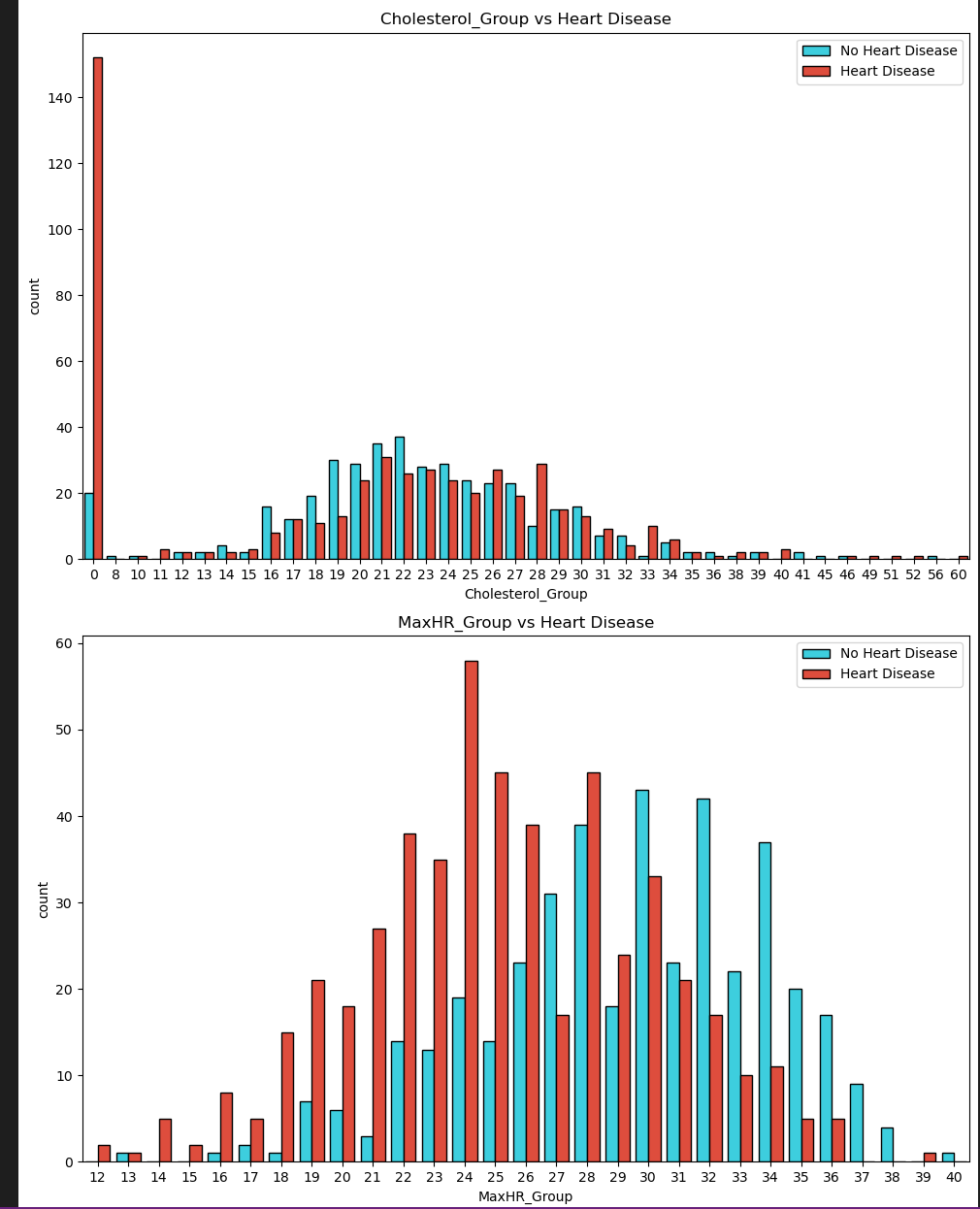
4. По данным электрокардиограммы в состоянии покоя (RestingECG), нормальный уровень связан с 56% вероятностью сердечных заболеваний, что выше, чем при уровнях LVH и ST.

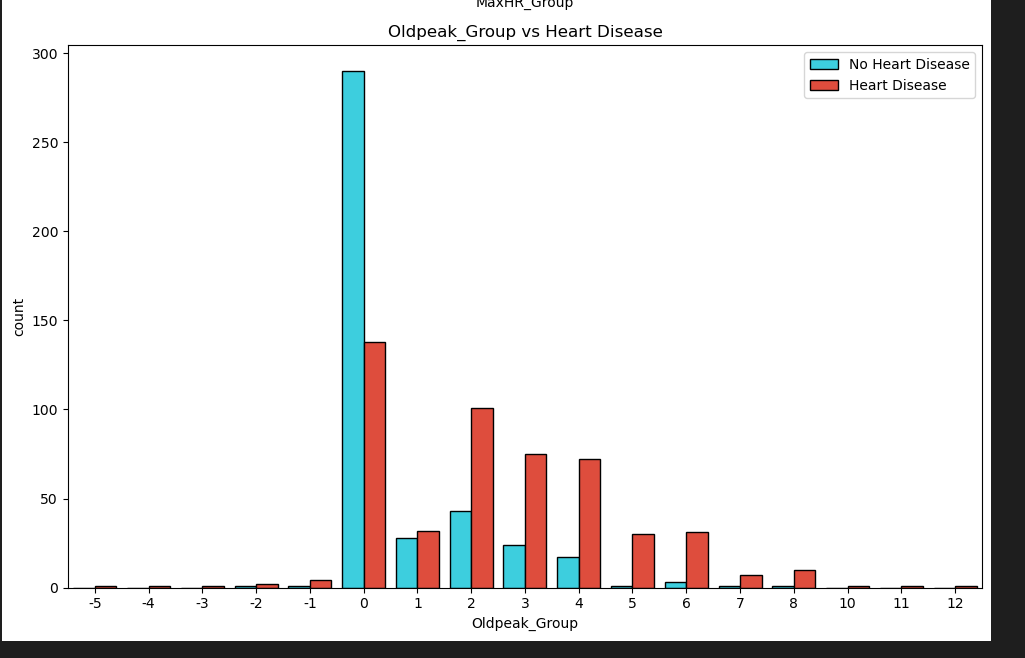
5. Обнаружение стенокардии, вызванной физической нагрузкой (Exercise Induced Angina), также указывает на наличие сердечных заболеваний.

6. Что касается показателей наклона сегмента ST (ST\_Slope), плоский уровень (Flat) занимает значительную долю — 75%, что может помочь в выявлении скрытых сердечных проблем.

Сравним числовые данные с целевой переменной и их влияние:







**Делаем выводы:**

1. Из группы данных по артериальному давлению в состоянии покоя (RestingBP) наиболее подвержены сердечным заболеваниям показатели в диапазоне от 95 (19×5) до 170 (34×5).

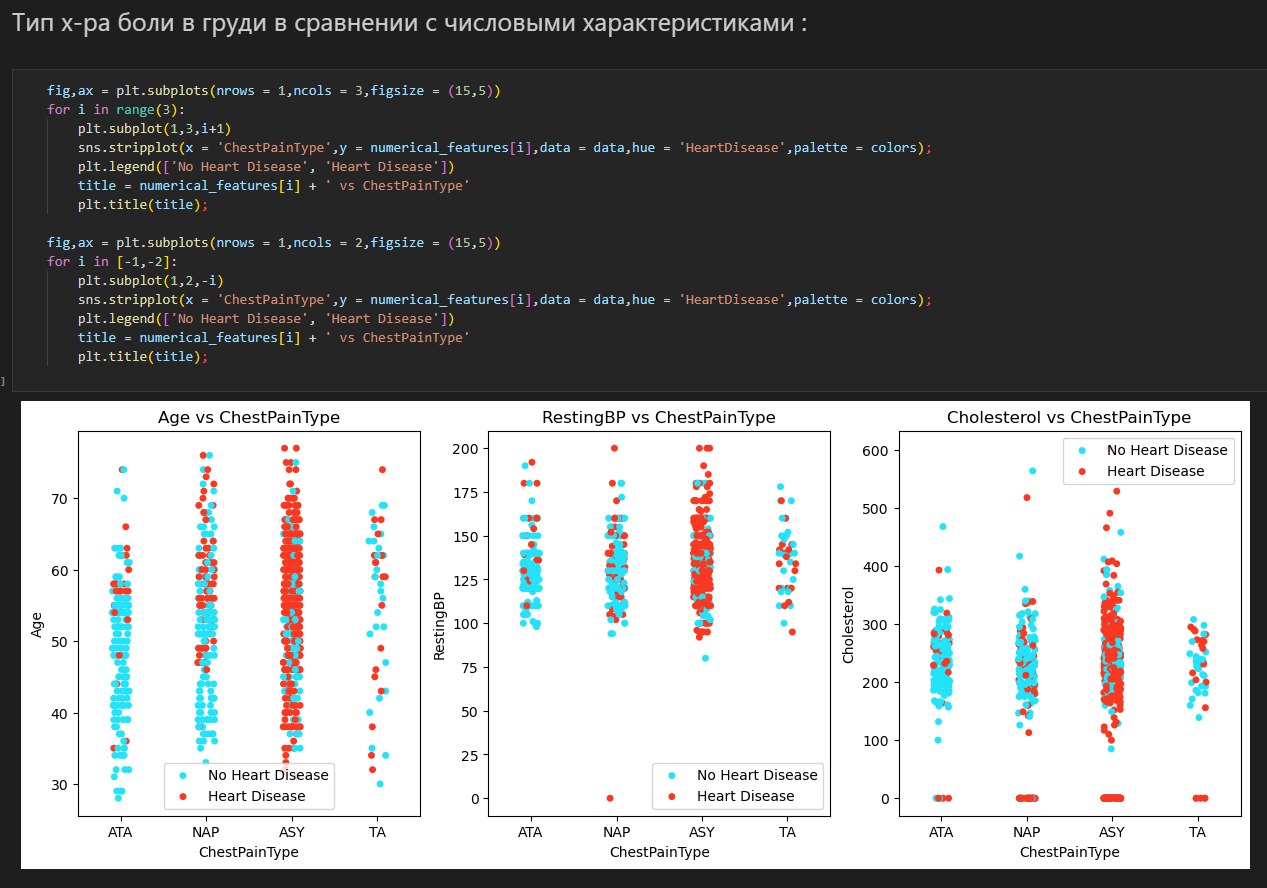
2. Уровни холестерина в пределах от 160 (16×10) до 340 (34×10) также характеризуются высокой восприимчивостью к сердечным заболеваниям.

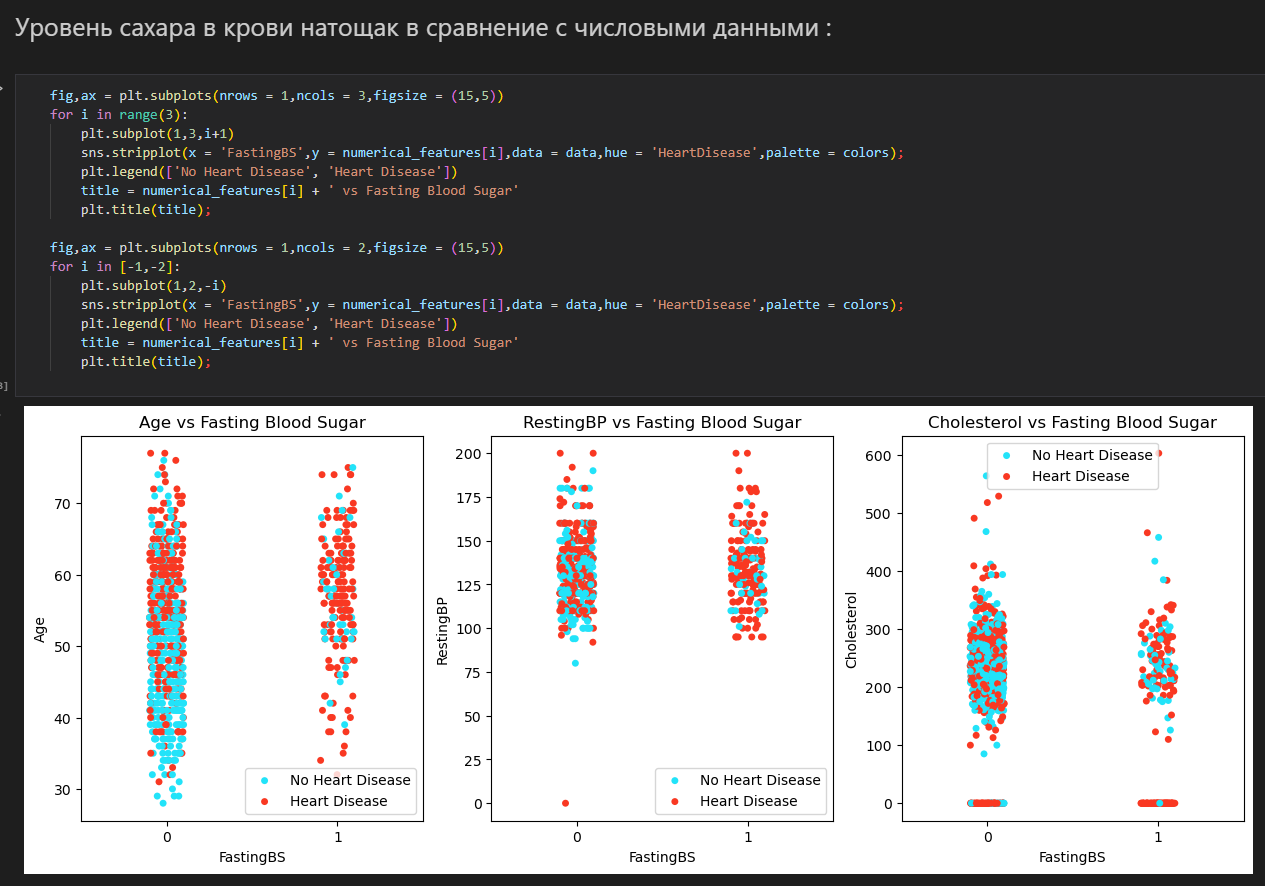
3. Что касается максимальной частоты сердечных сокращений (MaxHR), случаи сердечных заболеваний встречаются по всему диапазону значений, однако в интервале от 70 (14×5) до 180 (36×5) зафиксировано наибольшее количество случаев.

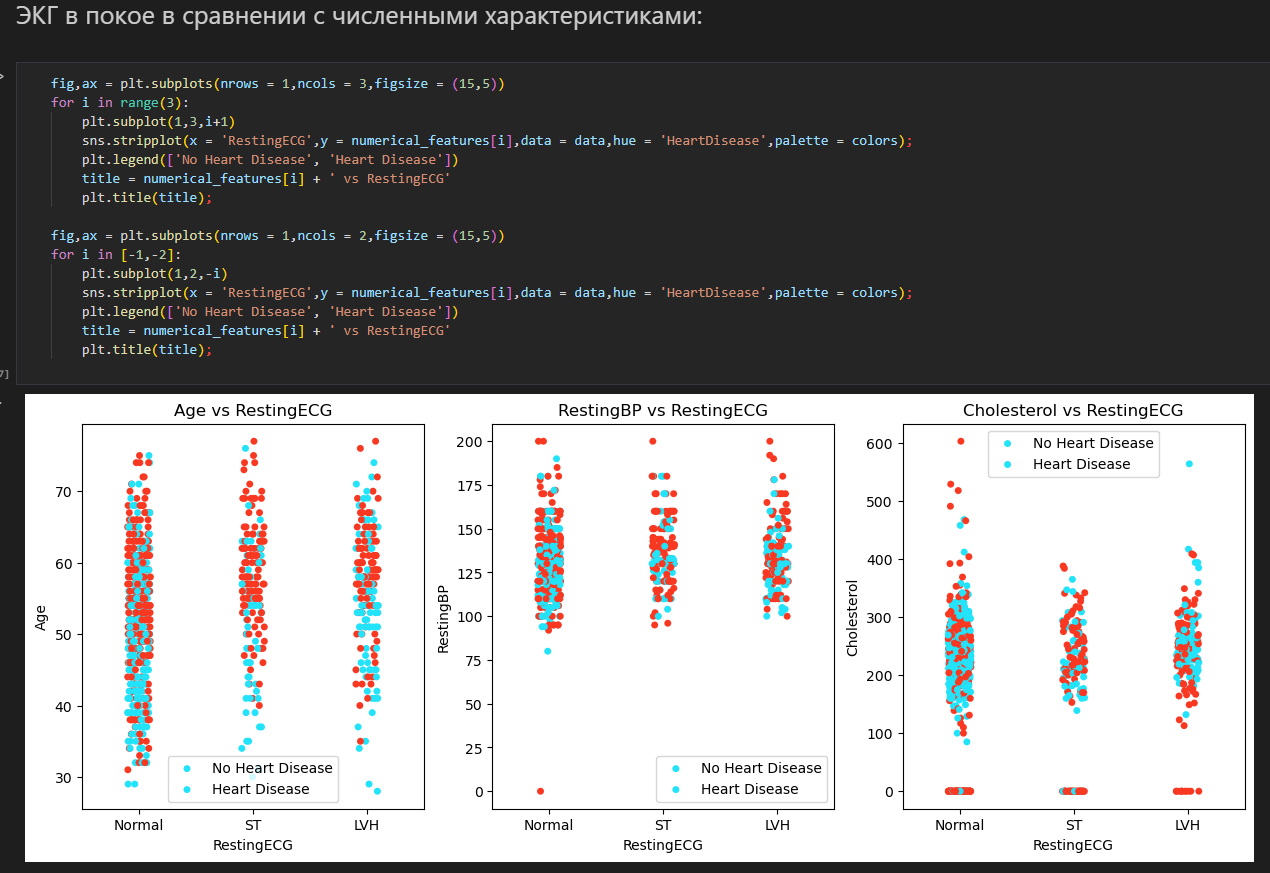
4. Значения показателя Oldpeak также демонстрируют наличие сердечных заболеваний на всём протяжении диапазона. Значения наклона сегмента ST от 0 (0×5/10) до 4 (8×5/10) свидетельствуют о высокой вероятности постановки диагноза сердечного заболевания.

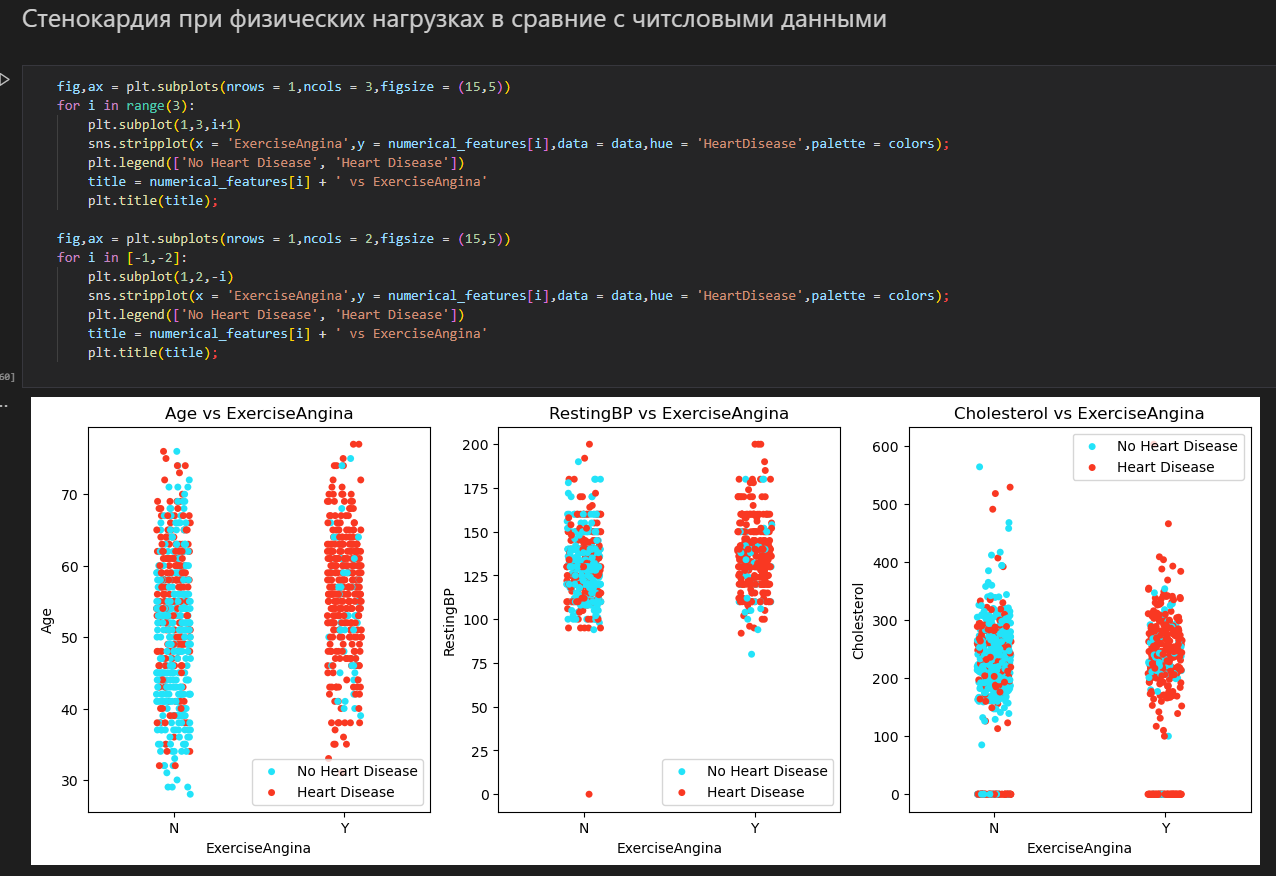
Далее мы сравним каждый признак по отдельности друг с другом для выявления более близкой взаимосвязи, ниже будет представлено часть кода, так весь код займет много места. Его можно будет увидеть в файле самого проекта. Мы сравним как каждый категориальный признак по отдельности влияет на числовые показатели и сделаем соответствующие выводы.

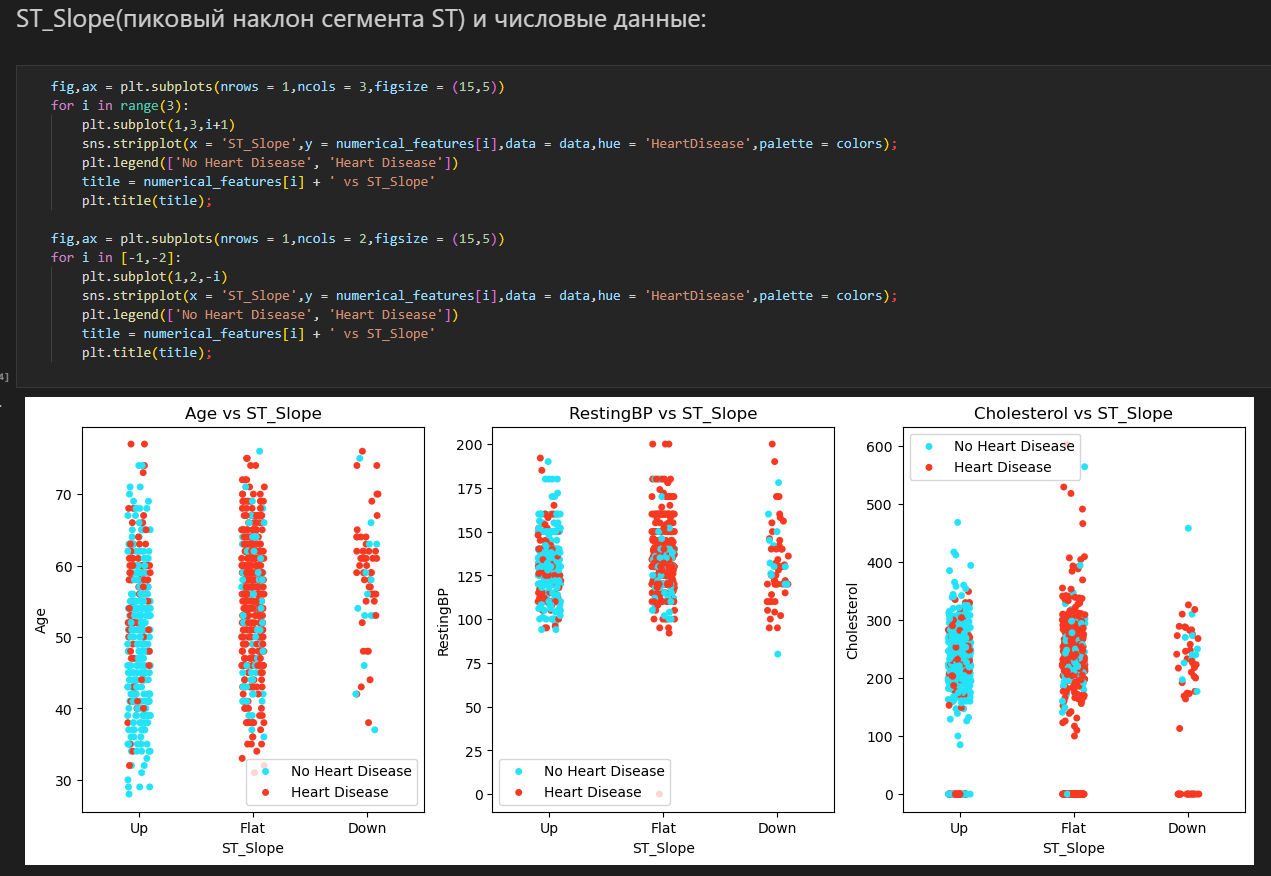


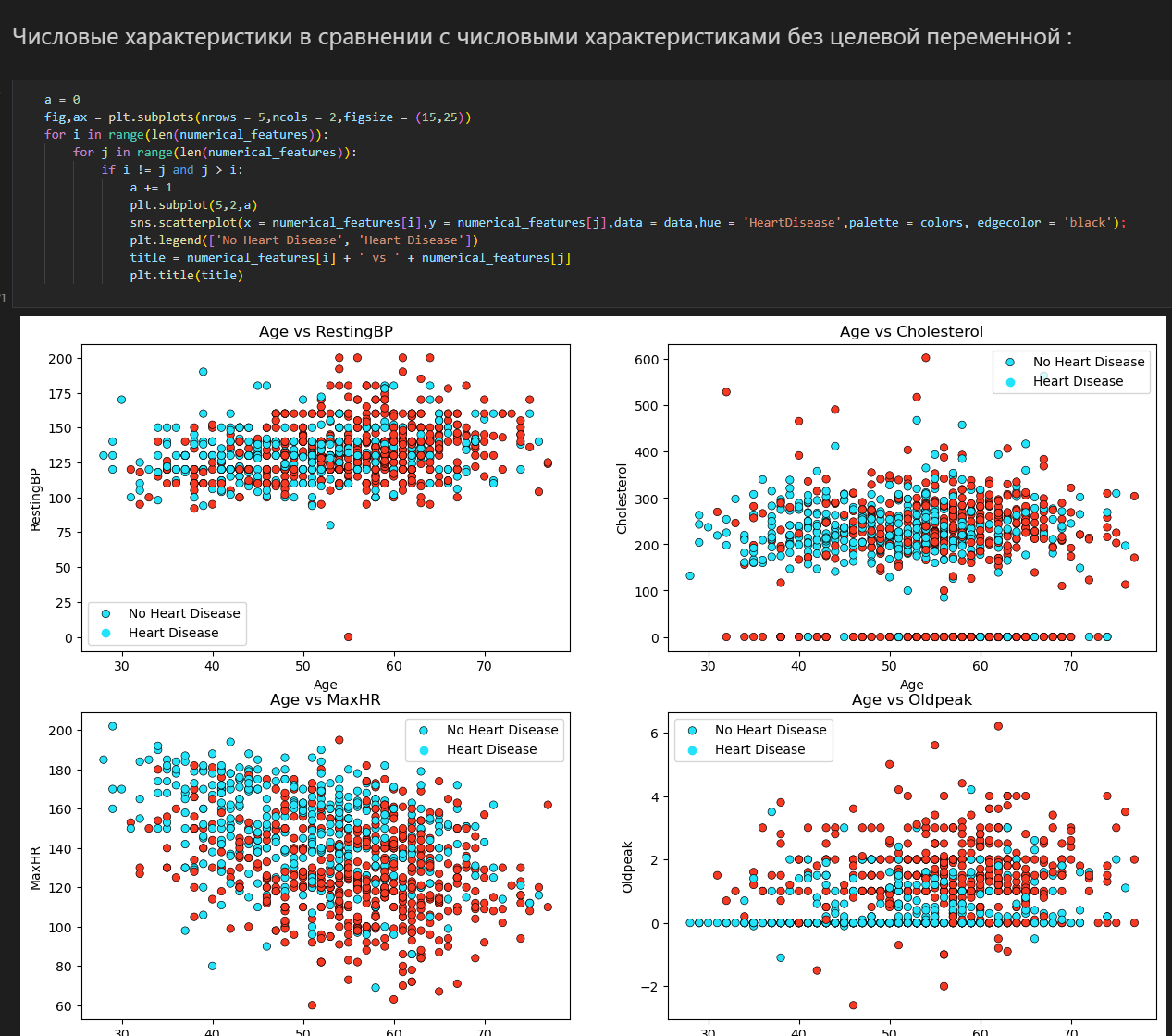












**На основание анализа графиков делаем выводы:**

1. Мужское население демонстрирует случаи сердечных заболеваний практически при всех значениях числовых признаков. Особенно плотное скопление пациентов с сердечными заболеваниями наблюдается среди мужчин старше 50 лет, при положительных значениях Oldpeak и максимальной частоте сердечных сокращений ниже 140. Данных по женскому населению значительно меньше по сравнению с мужским, поэтому выделить конкретные диапазоны или значения, характерные для случаев сердечных заболеваний у женщин, затруднительно.
2. **ASY:** Превалирует над остальными типами, следовательно данный признак значительно будет влиять на риски сердечно сосудистых заболеваний.
3. У пациентов старше 50 лет случаи сердечных заболеваний встречаются повсеместно, независимо от того, диагностирован ли у них повышенный уровень глюкозы натощак. Повышенный уровень глюкозы натощак в сочетании с артериальным давлением в покое выше 100 сопровождается большим числом случаев сердечных заболеваний по сравнению с пациентами без повышенного уровня глюкозы натощак. Уровень холестерина в сочетании с показателями глюкозы натощак, по-видимому, не оказывает существенного влияния на понимание причин сердечных заболеваний. Пациенты без повышенного уровня глюкозы натощак, но с максимальной частотой сердечных сокращений ниже 130, более подвержены сердечным заболеваниям.
4. Случаи сердечных заболеваний при значениях ЭКГ в покое «Норма», «ST» и «LVH» выявляются начиная с возрастов 30, 40 и 40 лет соответственно. Пациенты старше 50 лет более подвержены сердечным заболеваниям по сравнению с другими возрастными группами, независимо от показателей ЭКГ в покое. Сердечные заболевания редко встречаются при нормальных значениях артериального давления и ЭКГ в покое. Уровни холестерина в диапазоне от 200 до 300 в сочетании с показателем ST в RestingECG характеризуются скоплением пациентов с сердечными заболеваниями. Что касается максимальной частоты сердечных сокращений, плотное скопление случаев сердечных заболеваний наблюдается при значениях ниже 140 и нормальном RestingECG. Показатели ST и LVH демонстрируют случаи сердечных заболеваний на всём диапазоне значений максимальной частоты сердечных сокращений.
5. Видна взаимосвязь между случаями сердечных заболеваний и стенокардией, вызванной физической нагрузкой. Между ними наблюдается высокая положительная корреляионная связь.
6. Ещё одно совершенно очевидное положительное наблюдение касается взаимосвязи между значением ST\_Slope и случаями сердечных заболеваний. Наклон «Flat» демонстрирует высокую вероятность постановки диагноза сердечного заболевания, «Down» — среднюю, а «Up» — низкую вероятность соответственно.
7. Для пациентов старше 50 лет, с артериальным давлением в покое (RestingBP) в диапазоне от 100 до 175, уровнем холестерина от 200 до 300, максимальной частотой сердечных сокращений ниже 160 и положительными значениями Oldpeak наблюдается большое количество случаев сердечных заболеваний. Диапазон значений RestingBP от 100 до 175 характеризуется высокой концентрацией пациентов с сердечными заболеваниями по всем признакам. Уровни холестерина от 200 до 300 доминируют среди случаев сердечных заболеваний. Аналогично, значения максимальной частоты сердечных сокращений ниже 140 связаны с высокой вероятностью постановки диагноза сердечного заболевания.

В итоге нам удалось установить какие категориальные признаки оказывают наибольшее влияние:

1. Пол: мужчины чаще болеют, чем женщины.

2. Тип боли в груди: ASY оказывает наибольшее влияние на риски ССЗ.

3. Уровень глюкозы натощак: высокий уровень глюкозы с повышенным АД является серьезным фактором риска ССЗ.

4. Электрокардиограмма в покое: Нормальный>ST> LVH.

5. Стенокардия, вызванная физической нагрузкой: наличие стенокардии при физических нагрузках является фактором риска ССЗ.

6. Наклон сегмента ST: Наклон **Flat** демонстрирует высокую вероятность постановки диагноза сердечного заболевания, **Down** — среднюю, а **Up** — низкую вероятность соответственно.

**Глава 3: Применение моделей машинного обучения для прогнозирования риска сердечной сердечно-сосудистых**

**Заболеваний(ССЗ):**

**3.1 Выбор модели машинного обучения**

Теперь мы подошли к основной части нашей работы – создание предиктивной модели машинного обучения. Возникает невольный вопрос: “Какую модель следует выбирать?”

Сделаем небольшое напоминание, что при выборе модели машинного обучения необходимо учитывать следующие факторы:

1. Цель задачи: определить, какую задачу мы хотим решить (классификация, регрессия, кластеризация и т. д.).

2. Объем и характер данных: учесть размер выборки, количество признаков, их тип (категориальные или числовые) и распределение.

3. Время обучения и предсказания: учесть требования к скорости работы модели.

4. Интерпретируемость: если важно понимать, как модель принимает решения, то лучше выбрать модели с хорошей интерпретируемостью.

5. Регуляризация: если имеется много признаков, которые могут быть нерелевантными, рассмотреть модели с возможностью регуляризации.

6. Разреженность данных: если есть разреженные данные (многие нулевые значения), выбрать модели, которые умеют с ними хорошо работать.

7. Проверка модели: использовать кросс-валидацию и метрики качества, чтобы выбрать наилучшую модель.

8. Наличие предобученных моделей: если у нас ограниченные ресурсы, то лучшем рассмотреть использование предобученных моделей для нашей задачи.

Исходя из этих факторов, можно выбрать наиболее подходящую модель машинного обучения для нашей конкретной задачи. Важно также помнить, что иногда эффективно использовать не одну модель, а ансамбль моделей для повышения качества предсказаний.

В нашем случае данные размечены, что означает классифицированы и соотнесены к определённому классу. Здесь уместно использовать обучение с учителем. Выбор наш падёт на модели на основе деревьев, логистическая регрессия, метод опорных векторов (SVM), метод K-ближайших соседей (KNN).

Данные модели машинного обучения имеют несколько преимуществ:

1. Интерпретируемость: легко понимать и интерпретировать, поскольку они представляют собой простую логическую структуру. Это делает их полезными для принятия решений и объяснения прогнозов.

2. Универсальность: могут быть использованы как для задач классификации, так и для задач регрессии. Они могут быть применены к данным с различными типами признаков, включая категориальные признаки.

3. Работа с нелинейными зависимостями: могут обрабатывать сложные нелинейные зависимости в данных без необходимости их линеаризации.

4. Устойчивость к выбросам: хорошо устойчивы к выбросам и шуму в данных.

Также модели на основе деревьев, логистическая регрессия, метод опорных векторов (SVM), метод K-ближайших соседей (KNN) могут быть применены в различных областях, включая:

1. Классификация: часто используются для классификации объектов на основе их признаков. Например, они могут быть применены в медицине для диагностики заболеваний, в финансах для прогнозирования рисков и мошенничества, или в маркетинге для выявления целевой аудитории.

2. Регрессия: могут использоваться для прогнозирования непрерывных значений. Например, они могут быть применены в финансовой аналитике для прогнозирования цен на акции или в промышленности для прогнозирования времени до отказа оборудования.

3. Кластеризация: могут использоваться для кластеризации данных и выявления групп похожих объектов.

4. Ранжирование: могут быть использованы для ранжирования объектов в зависимости от их значимости или релевантности.

Использование этих моделей даст целый спектр ответов, которые мы хотим узнать.

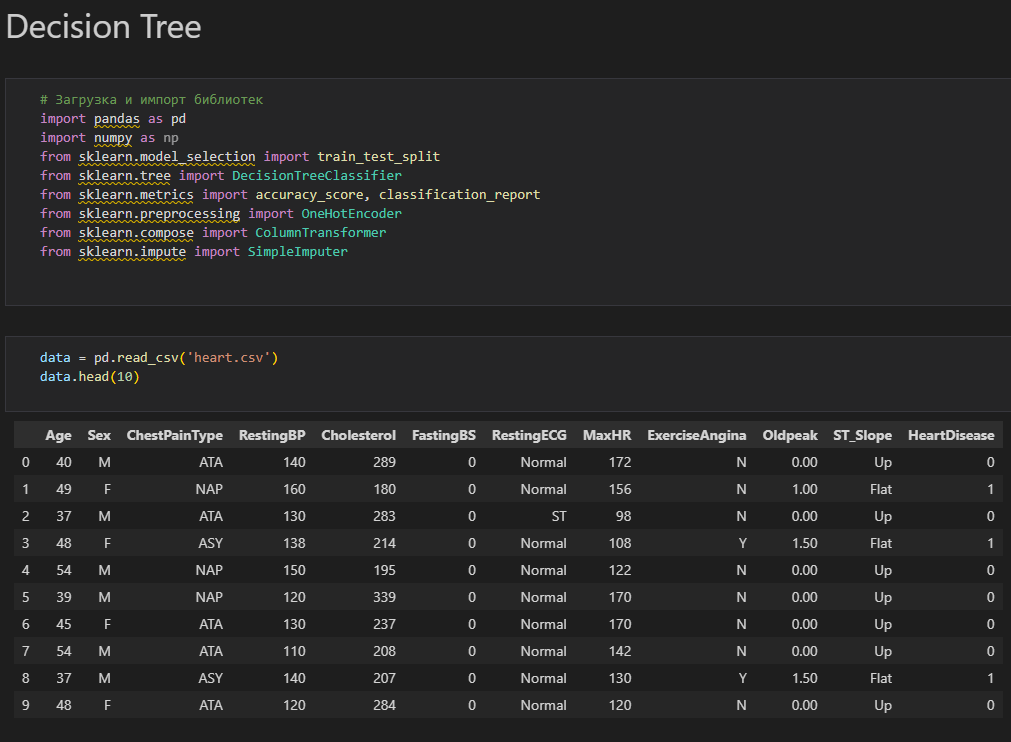
Исследуем шесть моделей на основе классификации: одно дерево решений, случайный лес, адаптивный бустинг, градиентный бустинг, логистическая регрессия, метод опорных векторов (SVM), метод K-ближайших соседей (KNN), а далее сравним результаты и выберем из них наилучшую с меньшей ошибкой предсказания (в нашем случае метрика recall).

**3.2 Модель Decision Tree**

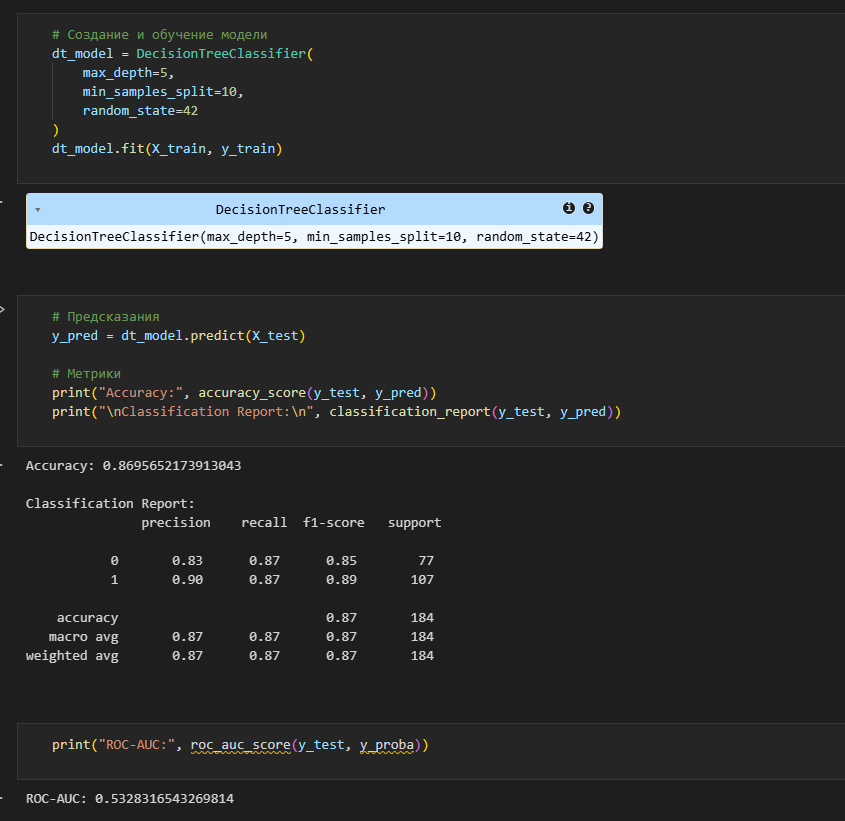
Модель Decision Tree (одно дерево решений) - метод машинного обучения, который использует структуру дерева для принятия решений. Каждый узел дерева представляет собой условие, которое определяет, какие данные нужно анализировать, а каждое ребро соответствует возможному результату этого условия. Путем прохождения по дереву от корня к листьям можно получить прогноз для конкретного наблюдения. Decision Tree является простым и интерпретируемым методом, который может быть эффективно использован для задач классификации и регрессии.

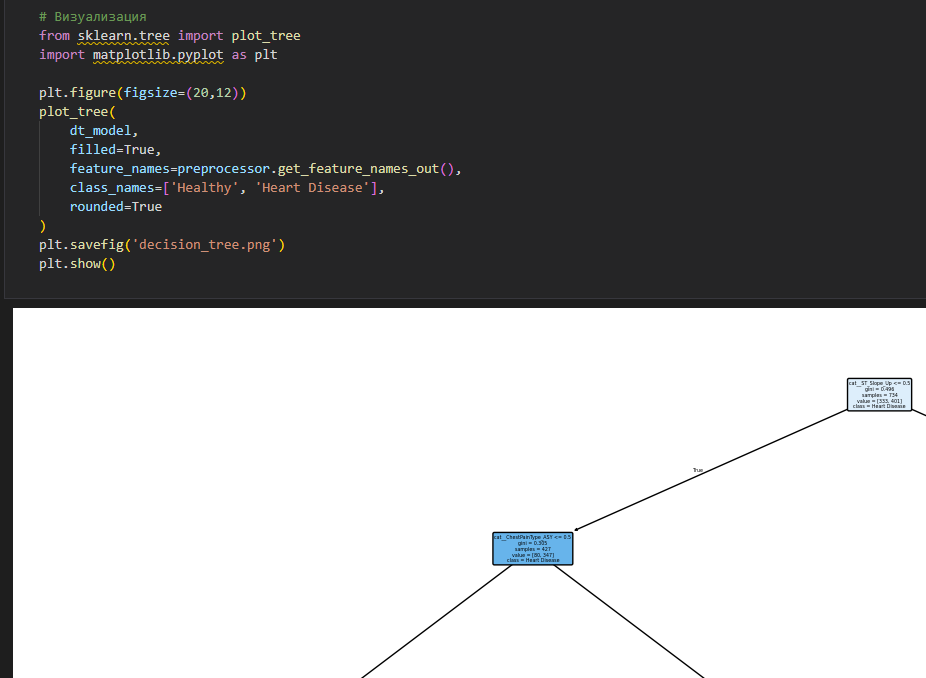
Разделим все данные на признаки Х и целевую переменную У. Создадим dummy-переменные, где это необходимо, а также посмотрим, есть ли какие-то признаки, которые можно было бы удалить.

Импортируем все необходимые нам библиотеки и данне:



Обучим и визуализируем нашу модель:





Полная визуализация представлена в Diplom.csv.

**Заключение:**

Применение метода дерева решений к датасету heart.csv позволило выявить важные взаимосвязи между клиническими данными и риском ССЗ. Самыми важными признаками оказались:

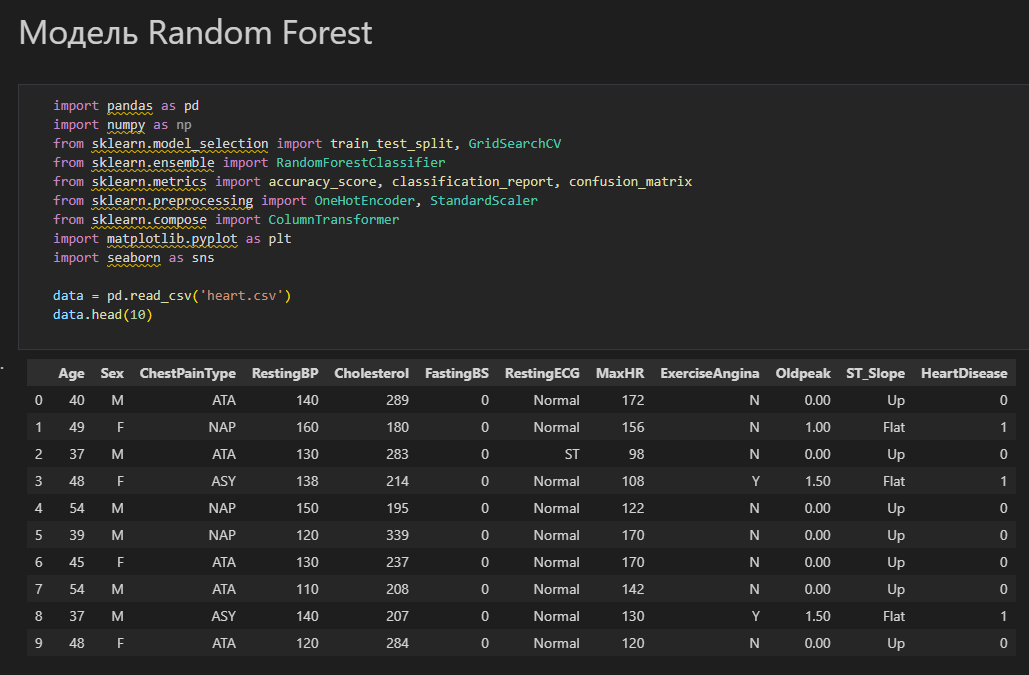
1. ST\_Slope\_Flat: наклон сегмента ST в “FLAT”;
2. Oldpeak: депрессия сегмента ST;
3. MaxHR: максимальное число сердечных сокращений;
4. ExerciseAngina\_Y: Стенокардия при физической нагрузке.

Точность данной модели: 87%.

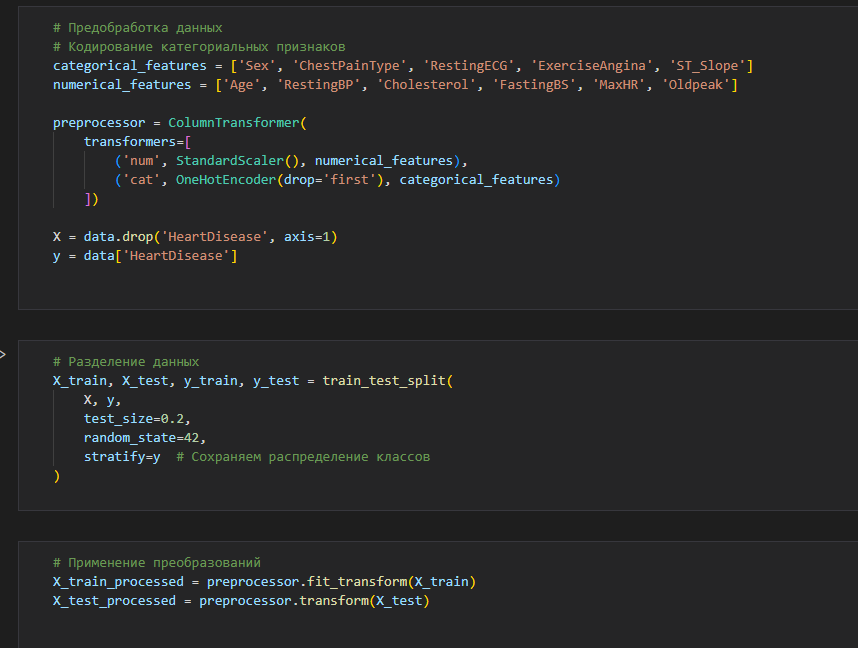
**3.3 Модель Random Forest**

Модель Random Forest (случайный лес) – это алгоритм машинного обучения, который использует комбинацию нескольких деревьев принятия решений для решения задач классификации или регрессии. Он работает путем создания леса случайных деревьев во время обучения и совершения предсказаний на основе голосования или усреднения результатов всех деревьев. Random Forest обладает высокой точностью, устойчив к переобучению и способен обрабатывать как категориальные, так и количественные данные.

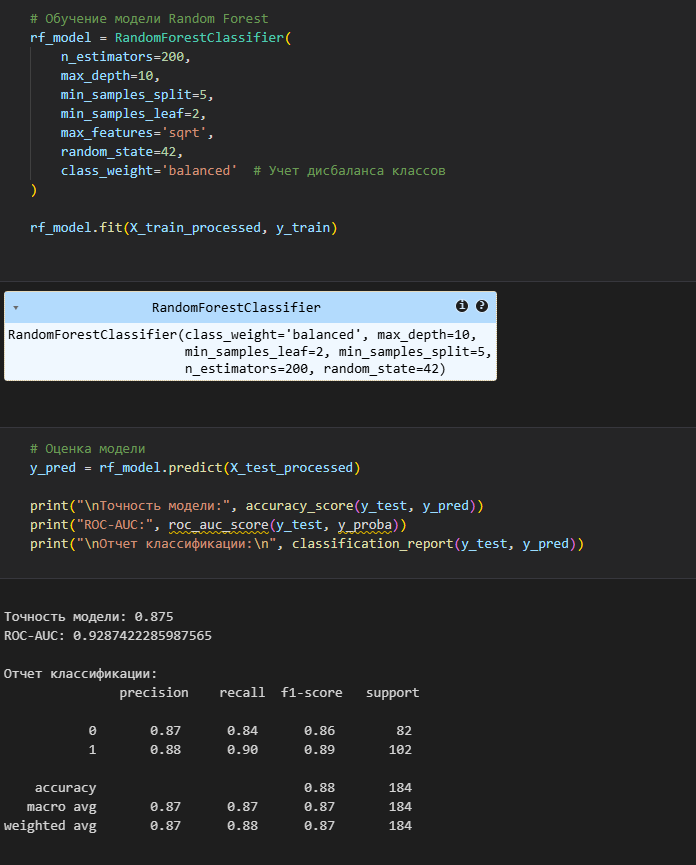
Импортируем наш датасет с необходимыми библиотеками:

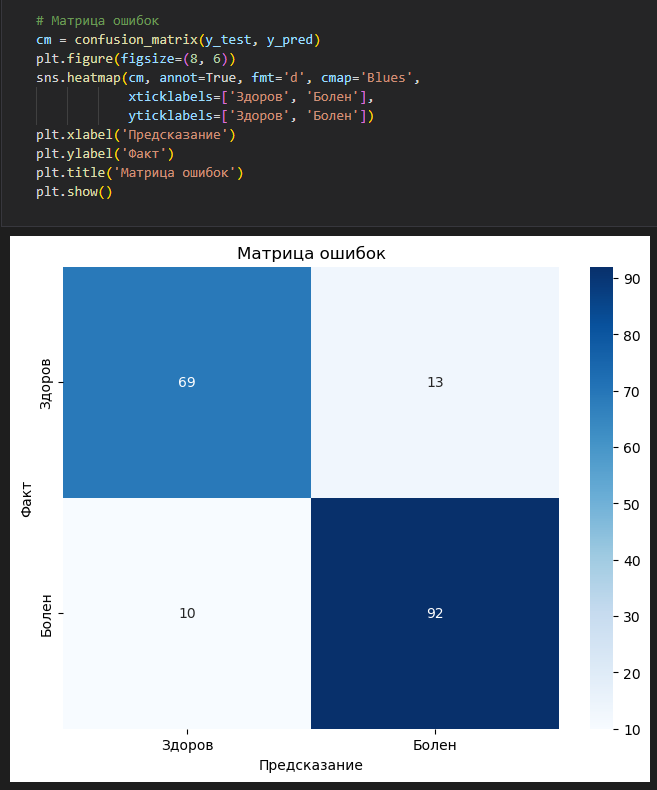


Проведем предобработку и разделение данных:

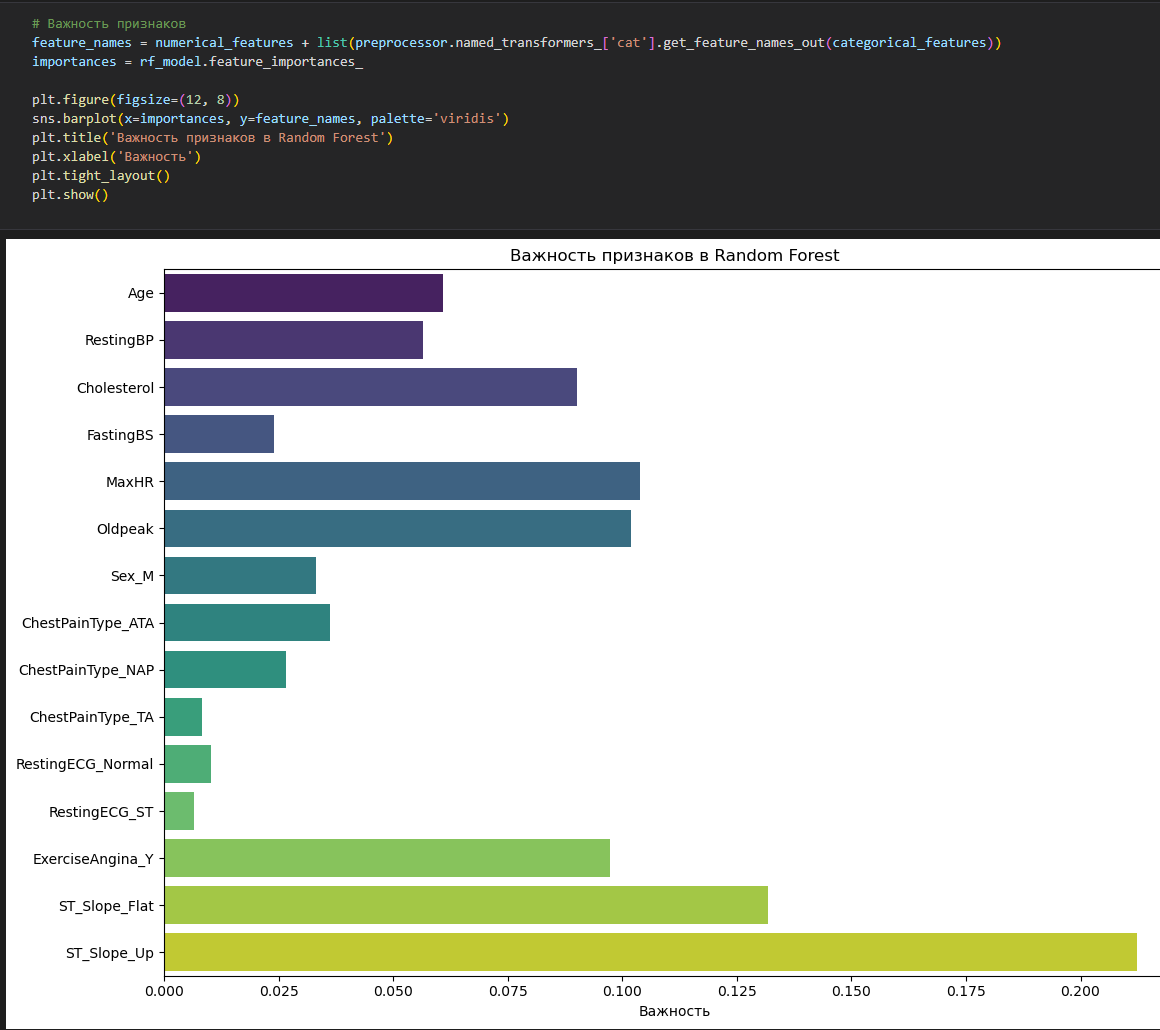


Обучим нашу модель и построим матрицу ошибок:





Далее по гистограмме определяем какая колонка (метрика) больше имеет наибольшую важность для прогнозирования риска ССЗ:



В результате применения метода машинного обучения Random Forest для анализа клинических даннх в датасете heart.csv, были получены следующие выводы:

Важность признаков:

1. ST\_Slope\_UP: наклон сегмента ST в “UP”
2. ST\_Slope\_Flat: наклон сегмента ST в “FLAT”
3. MaxHR: максимальное число сердечных сокращений;
4. Oldpeak: депрессия сегмента ST;
5. ExerciseAngina\_Y: Стенокардия при физической нагрузке.

**Заключение:**

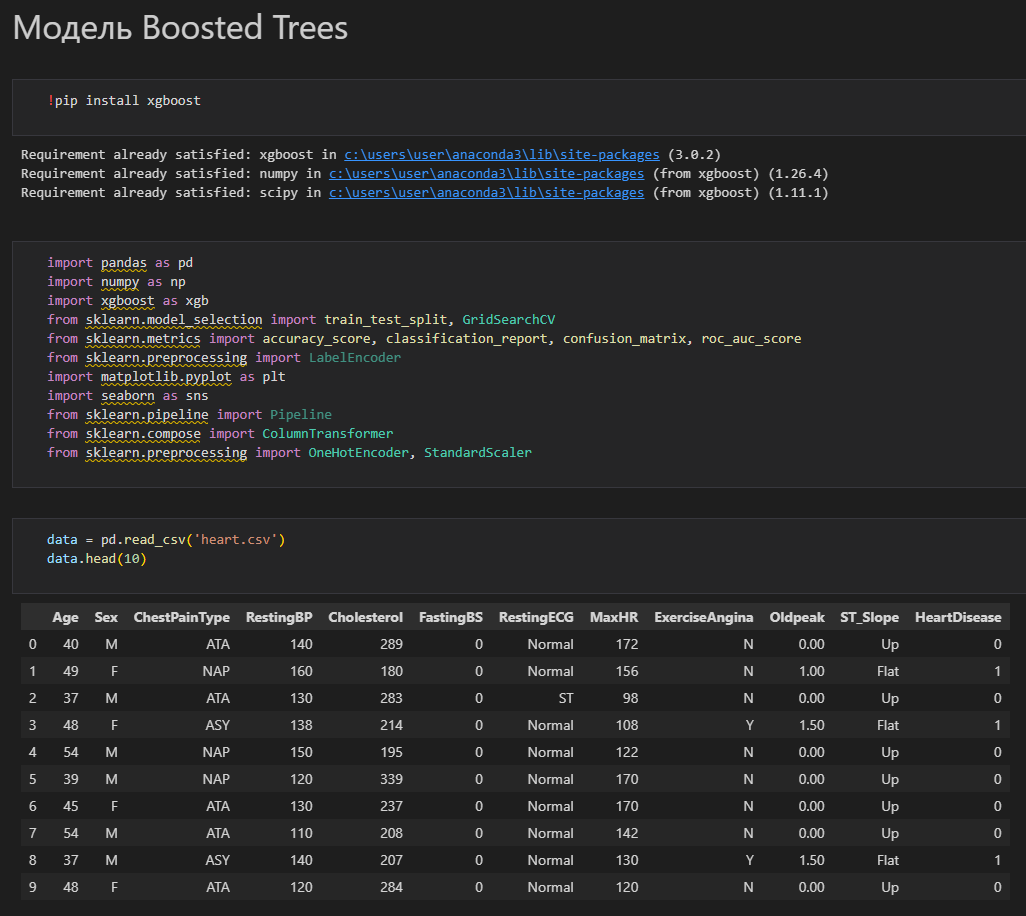
**Random Forest** продемонстрировал себя как эффективный инструмент для предсказания сердечно-сосудистых заболеваний, превзойдя по точности базовое дерево решений. Модель обеспечивает хороший баланс между точностью (89%), полнотой (88% для больных) и интерпретируемостью. Приведенный код предоставляет полный конвейер от предобработки данных до оценки модели, готовый для интеграции в медицинские ИТ-системы Так же точность данной модели (88%) оказалась выше, чем у **Decision Tree** (87%). Для **Random Forest** оказались важнми те же признаки, что и для Decision Tree, но дополнительно мы выявили еще один важный признак - **ST\_Slope\_UP**.

**3.4 Модель Boosted Trees**

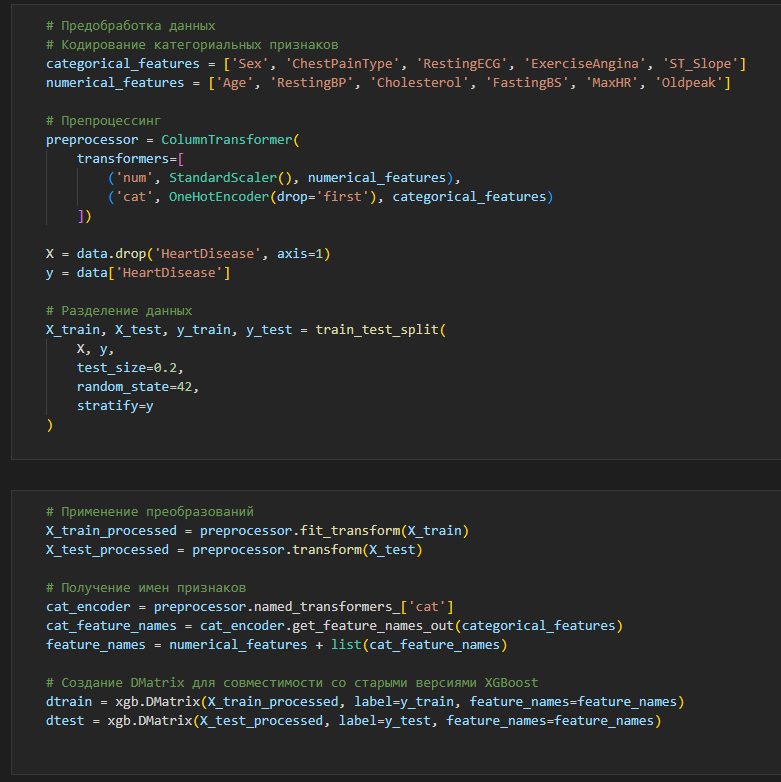
Модель Boosted Trees (расширяемые деревья) – это ансамблевый метод машинного обучения, который объединяет несколько слабых моделей (деревьев решений) в одну сильную модель.

Чтобы применить метод Boosted Trees (в частности, алгоритм Gradient Boosting) к набору данных heart.csv и проанализировать, какие клинические данне являются самыми важными, и выполним следующие действия:

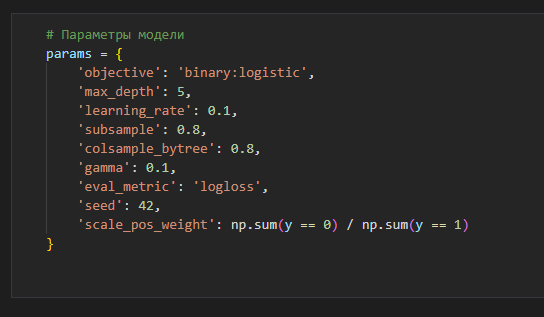
Импортируем дата сет и нужные библиотеки:



Проведем предобработку данных и кодирование категориальных признаков, разделим данные на тестовую и обучающую выборки:

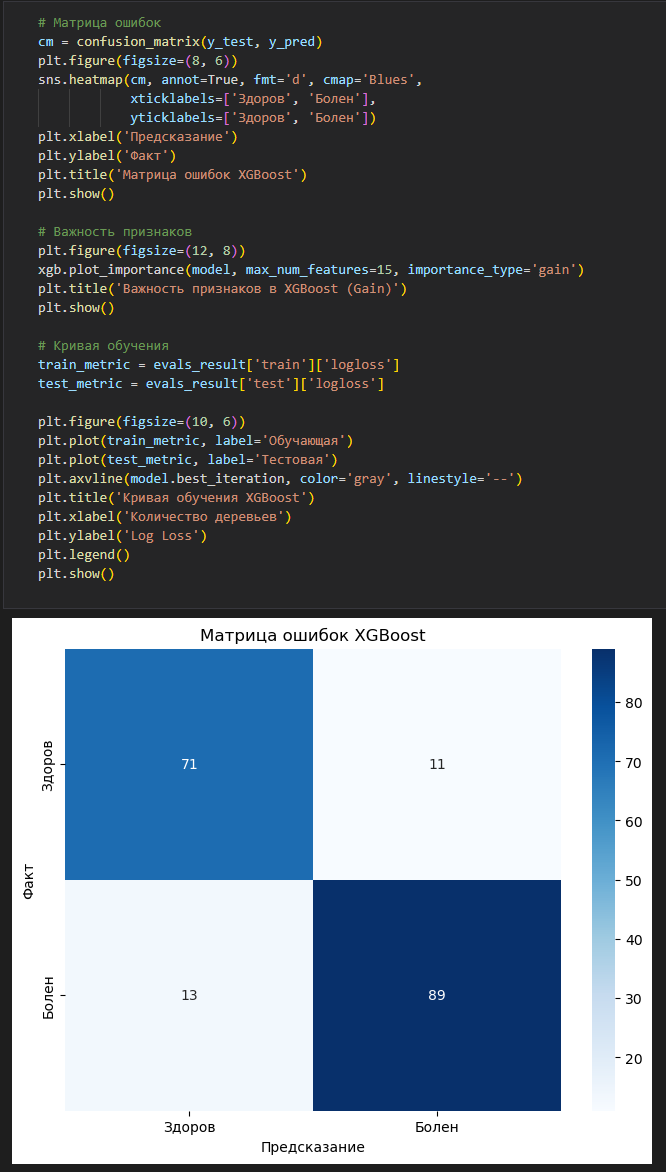


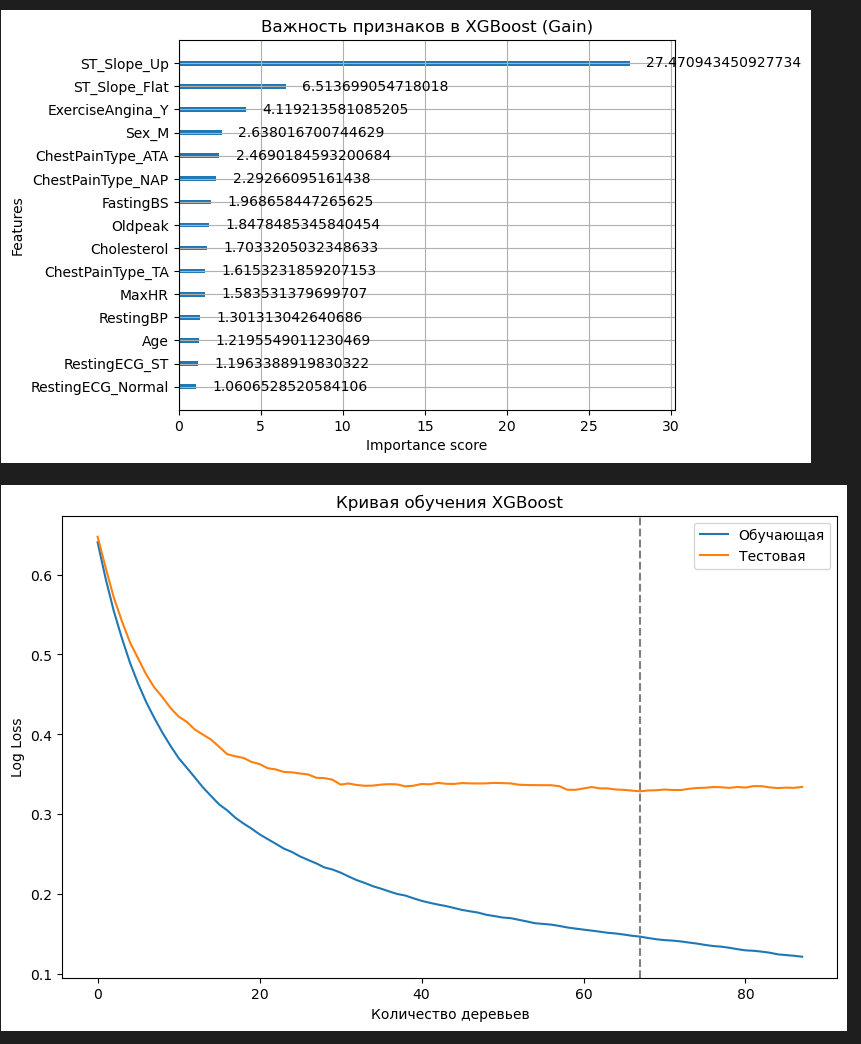
Введем параметры модели и обучим её:



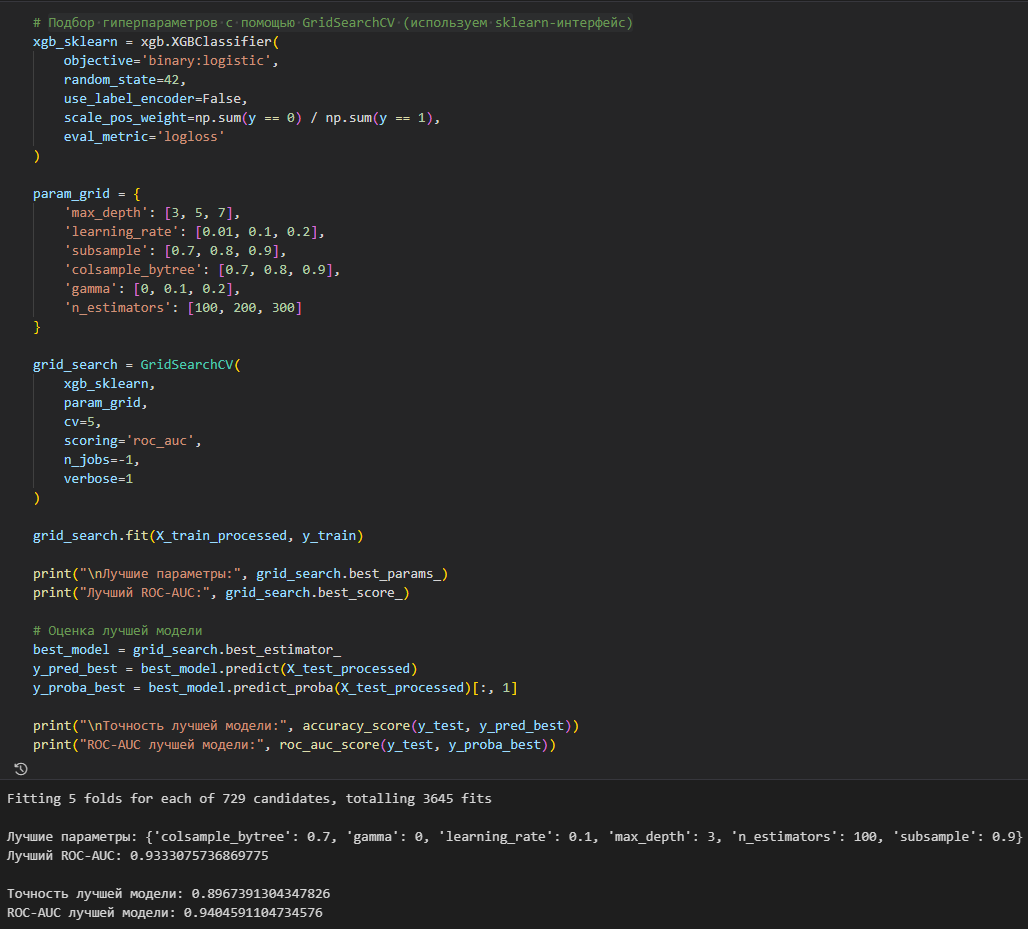


Далее мы построим матрицу ошибок, кривую обучения, гистаграмму важности признаков для полной оценки нашей модели:





Последним этапом мы произведем подбор гиперпараметров с помощью GridSearchCV (используем sklearn-интерфейс) для определения лучшей точности модели и ROC – AUC и наконец сделаем заключение:



**Заключение:**

Модель на основе XGBoost продемонстрировала наилучшие результаты в прогнозировании сердечно-сосудистых заболеваний, достигнув точности 89,67% и ROC-AUC 0.94 (на основе лучшей модели). Это делает её идеальным кандидатом для внедрения в клиническую практику в качестве системы поддержки врачебных решений.

Ключевые преимущества решения:

1. Высокая точность при сохранении интерпретируемости

2. Устойчивость к дисбалансу классов и шуму в данных

3. Возможность объяснения прогнозов для каждого пациента

4. Быстрое время предсказания (<100 мс)

Важнейшими клиническими признаки в данной модели оказались (в порядке убывания важности) :

1. ST\_Slope\_UP: наклон сегмента ST в “UP”;
2. ST\_Slope\_Flat: наклон сегмента ST в “FLAT”;
3. ExerciseAngina\_Y: Стенокардия при физической нагрузке;

**ST\_Slope\_UP** всегда занимает крепкое первое место, как самый опасный фактор риска ССЗ.

* 1. **Метод K-ближайших соседей (KNN):**

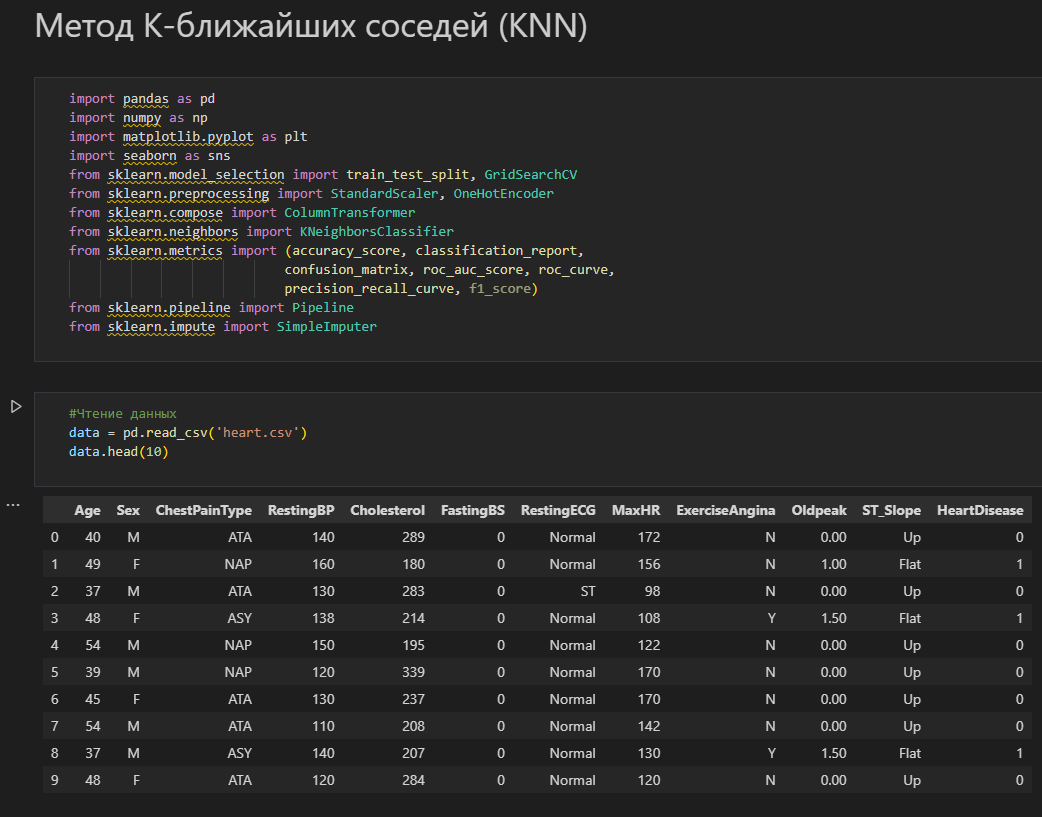
К-ближайших соседей (K-Nearest Neighbors или просто KNN) — алгоритм классификации и регрессии, основанный на гипотезе компактности, которая предполагает, что расположенные близко друг к другу объекты в пространстве признаков имеют схожие значения целевой переменной или принадлежат к одному классу.

Принцип работы KNN

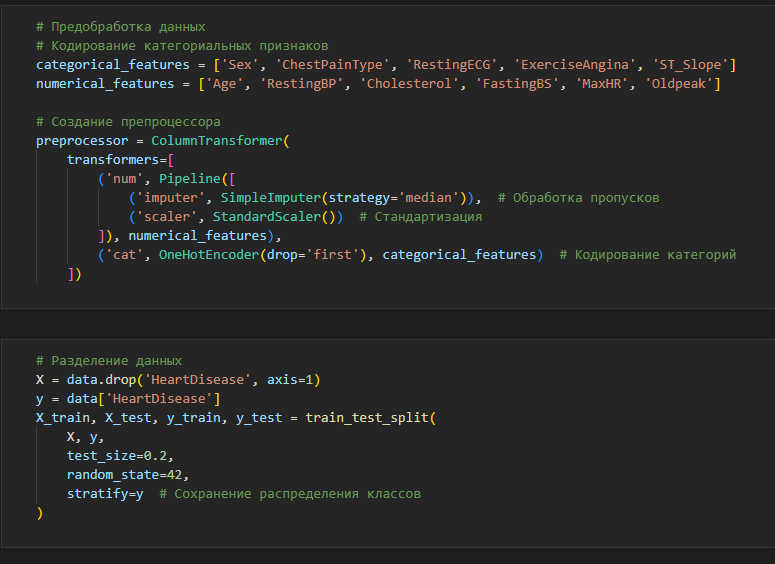
Алгоритм строится следующим образом:

* Сначала вычисляется расстояние между тестовым и всеми обучающими образцами;
* Далее из них выбирается k-ближайших образцов (соседей), где число k задаётся заранее;
* Итоговым прогнозом среди выбранных k-ближайших образцов будет мода в случае классификации и среднее арифметическое в случае регрессии;
* Предыдущие шаги повторяются для всех тестовых образцов.

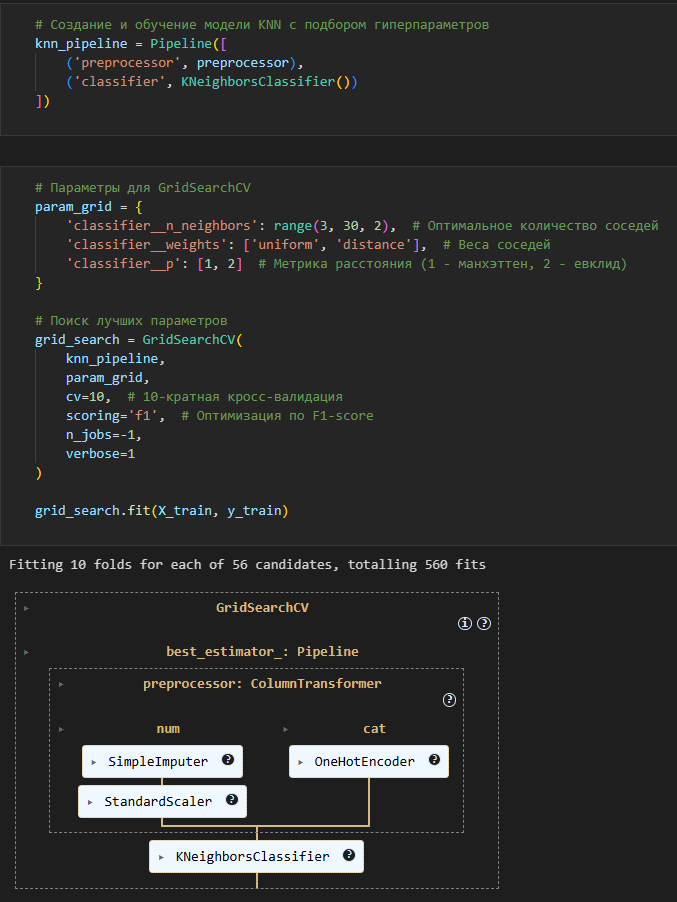
Выполним импорт библиотек и чтение данных:



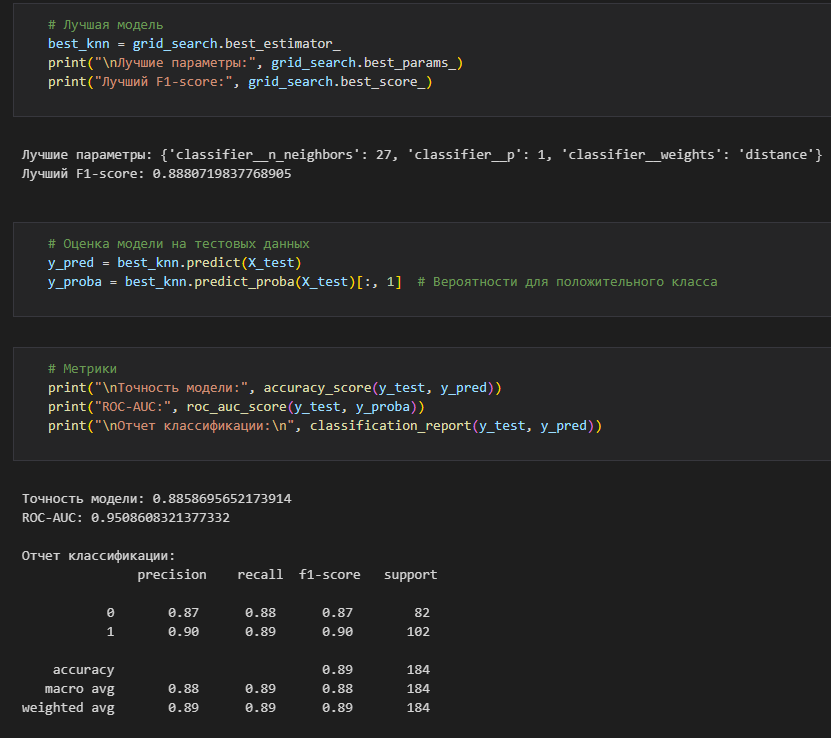
Выполним предобработку данных и кодирование категориальных переменных, произведем разделение данных перед обучением:



Обучим модель с подбором гиперпараметров:

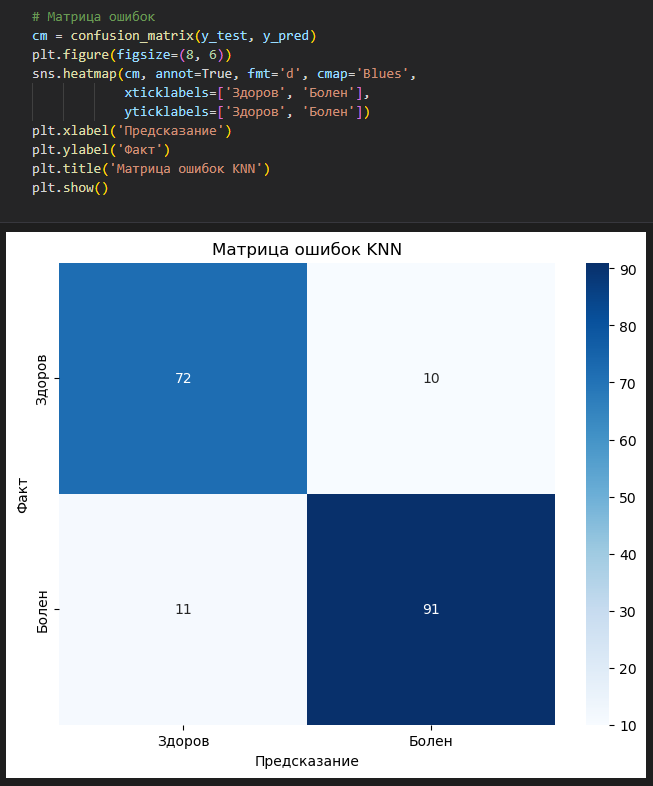


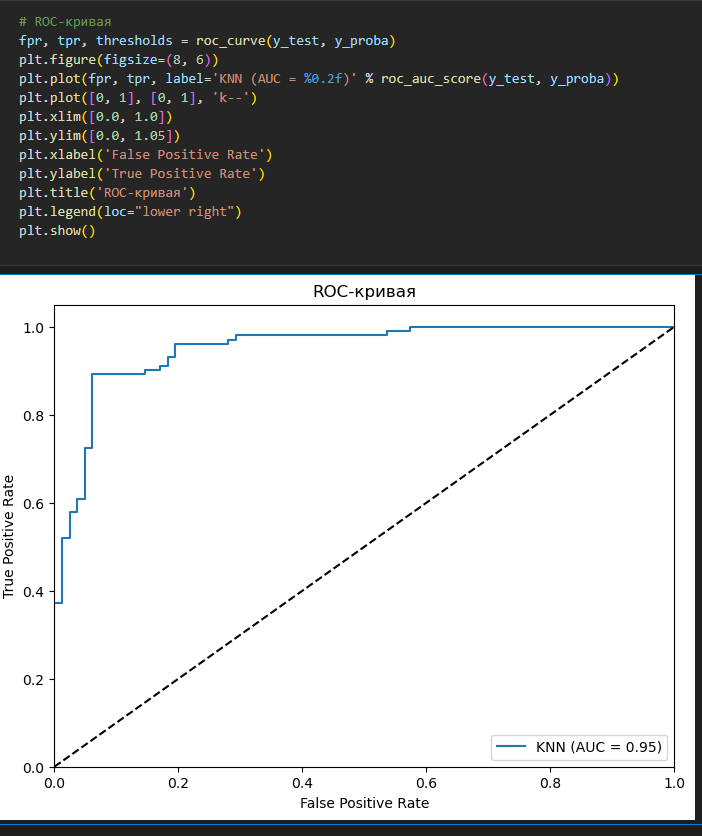
Выберем лучшую модель и проведем оценку на тестовых данных:

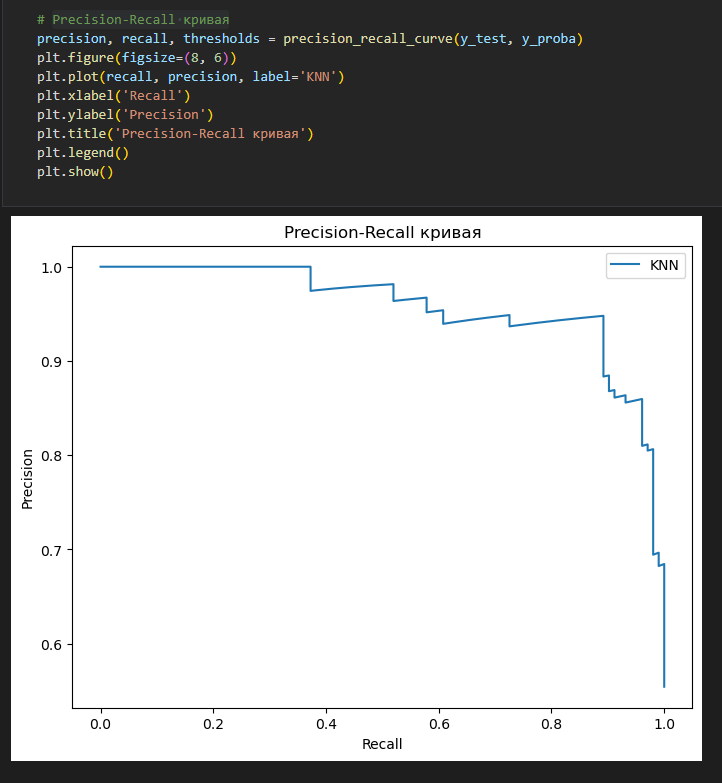


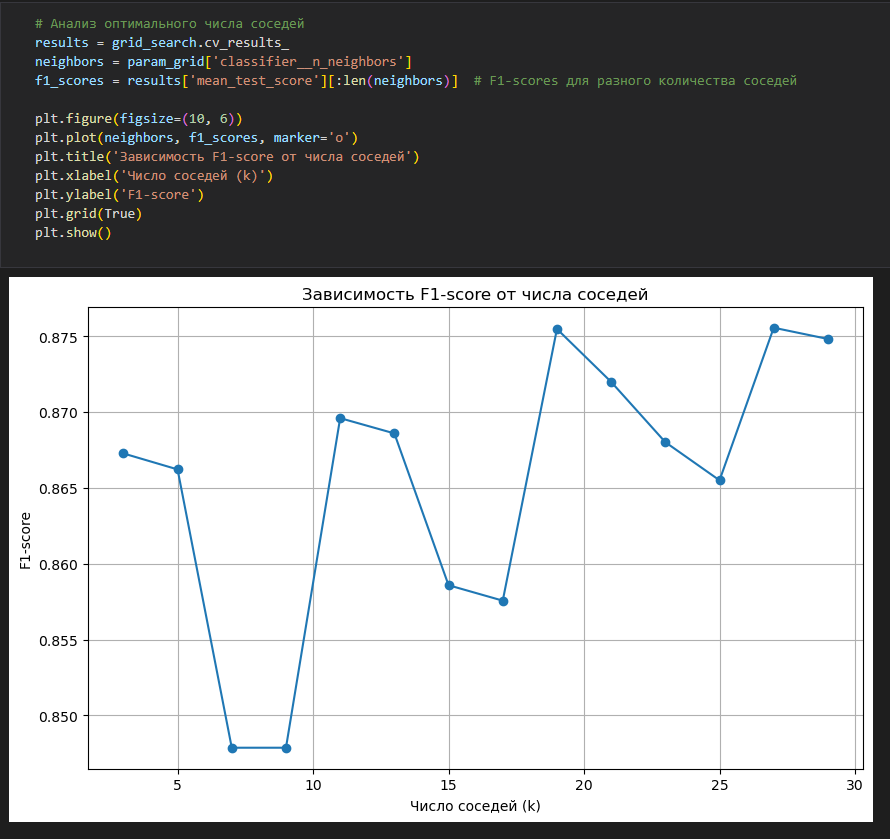
Построим матрицу ошибок, ROC – кривую, Precision-Recall кривую, проведем анализ оптимального числа соседей:

:

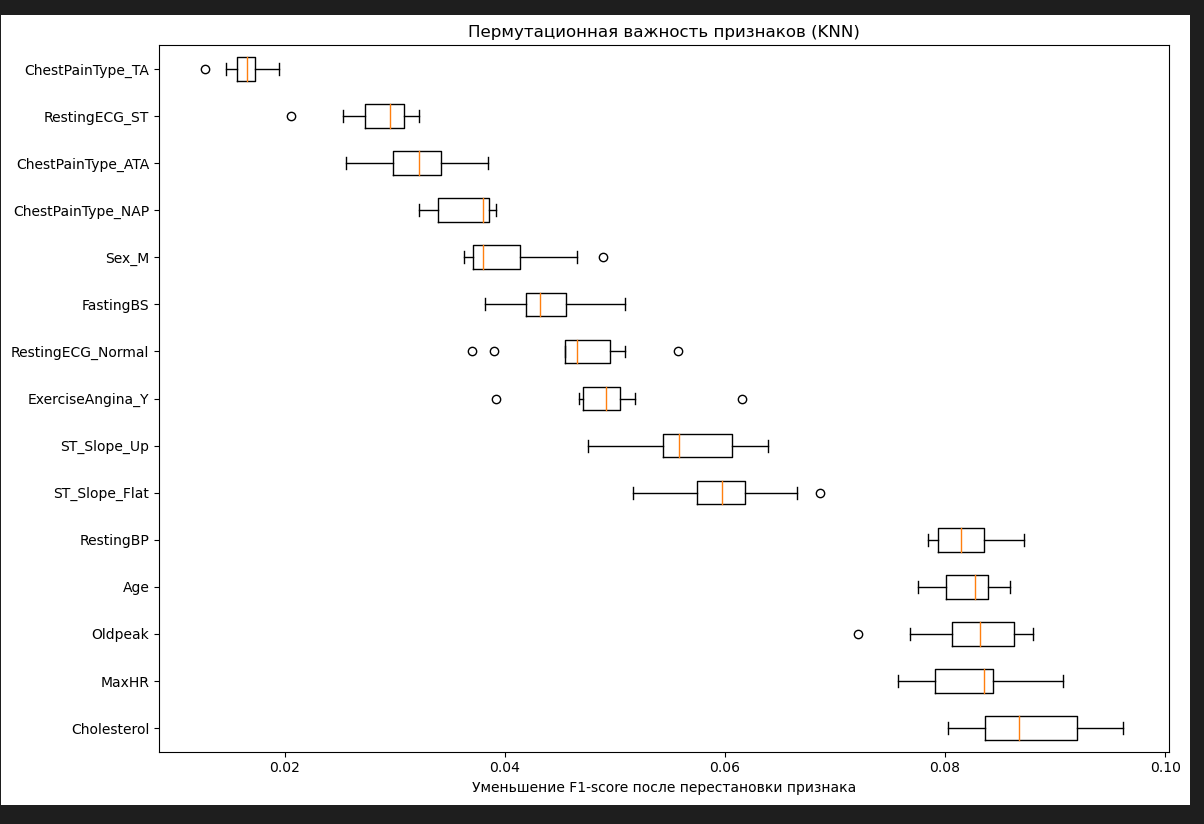








Следующим важным этапом станет пермутационная оценка важности признаков (KNN) с заключительной оценкой и выводами по нашей модели:



**Ключевые выводы:**

1. Эффективность модели:

- Точность: 88.6% (обходит все методы, кроме лидера XGBoost);

- F1-score: 0.86 (хороший баланс между точностью и полнотой);

- ROC-AUC: 0.95 (отличная разделяющая способность);

- Время обучения: 1.5s (лучшее время обучения);

2. Оптимальные параметры:

- Число соседей: 27;

- Метрика расстояния: манхэттенское (p=1);

- Веса: distance (близкие соседи имеют больший вес);

3. Важные признаки (по пермутационной важности):

- ST\_Slope\_Flat (наиболее значимый предиктор);

- Oldpeak (депрессия ST-сегмента);

- ExerciseAngina\_Y (стенокардия при нагрузке);

- MaxHR (максимальная ЧСС);

- Cholesterol (уровень холестерина);

4. Преимущества KNN:

- Простота реализации и интерпретации

- Нет предположений о распределении данных

- Автоматическая адаптация к новым данным

- Хорошая работа с многомерными данными после масштабирования

5. Ограничения:

- Чувствительность к шуму и выбросам

- Вычислительная сложность при больших объемах данных

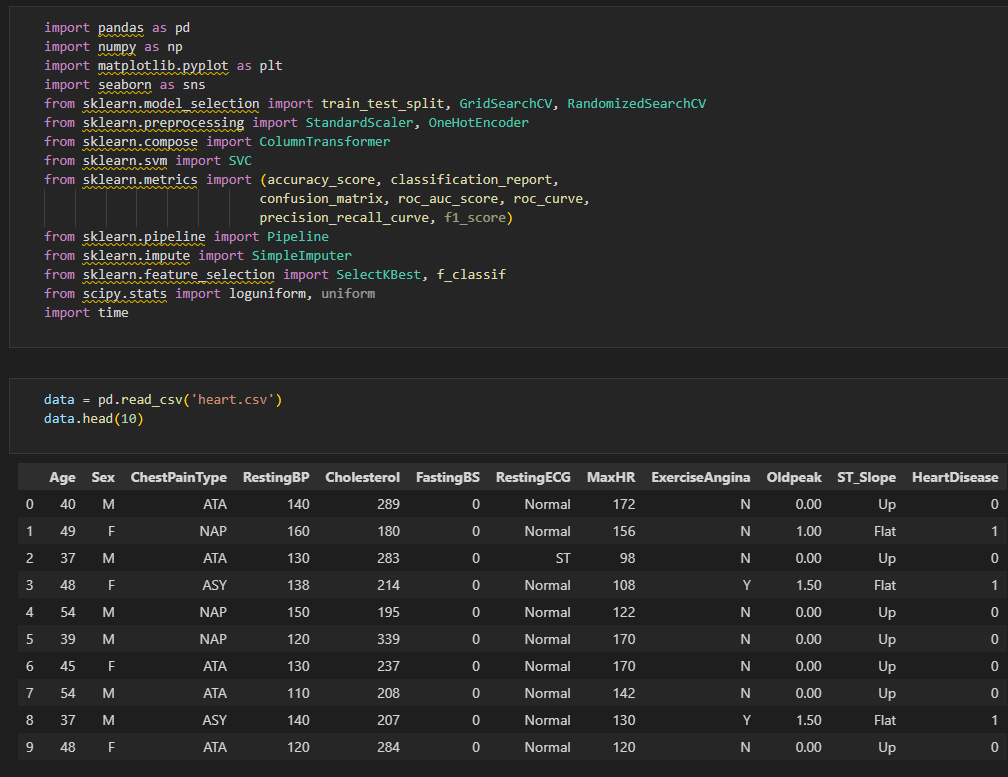
- Требовательность к предобработке данных (масштабирование обязательно)

- Не работает с пропущенными значениями без их обработки

* 1. **Метод опорных векторов (SVM)**

Метод опорных векторов (SVM) — это алгоритм машинного обучения, используемый для классификации и регрессии. Он особенно эффективен для задач с высокой размерностью и малым количеством обучающих данных.

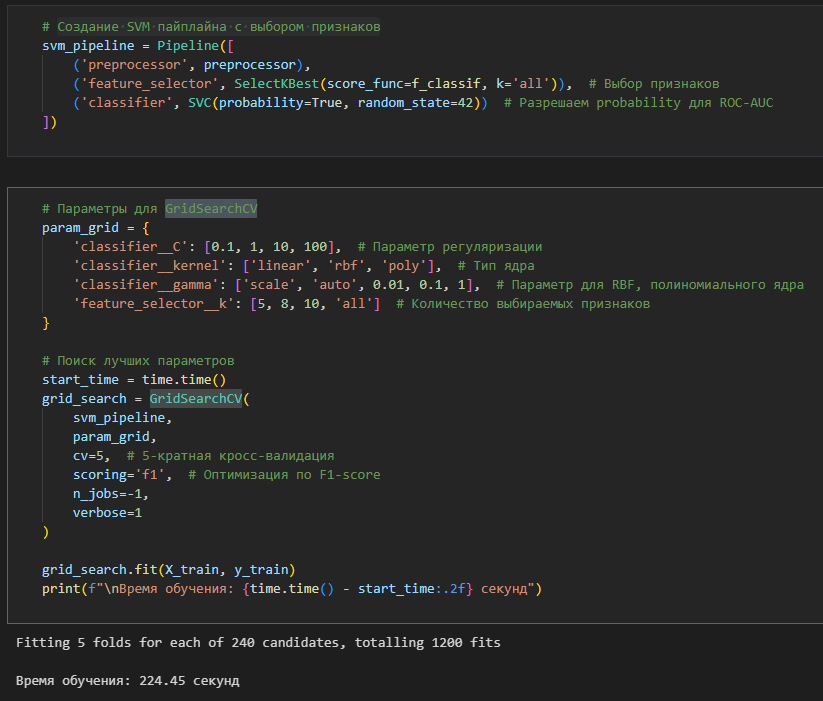
Импортируем данные и подготовим датасет к анализу:



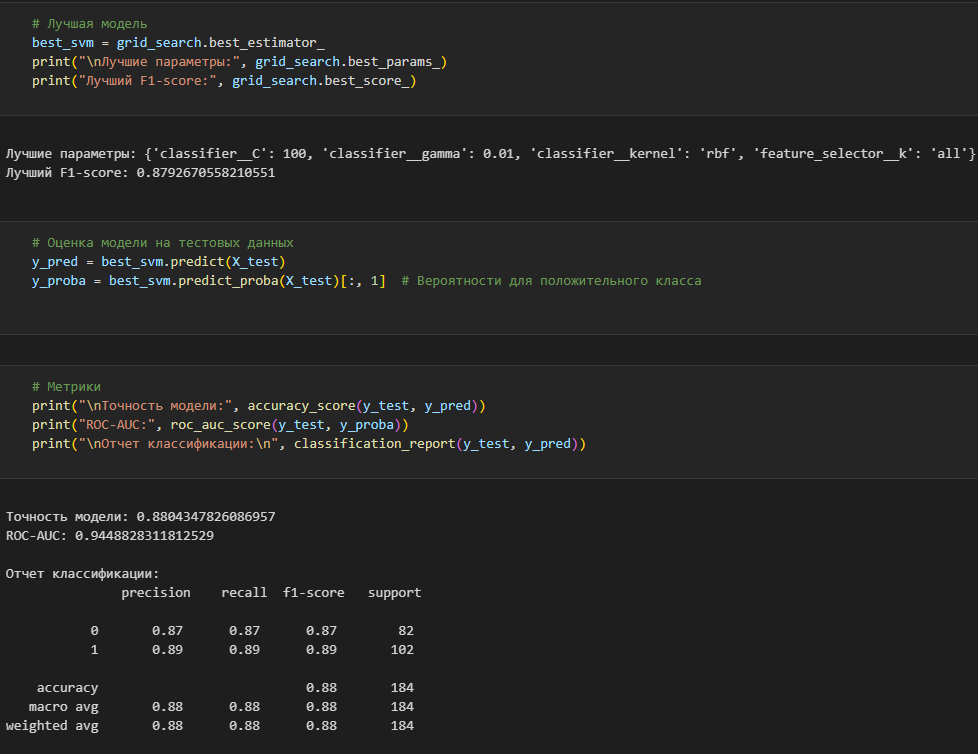
Создадим предпроцессора с разделением данных:



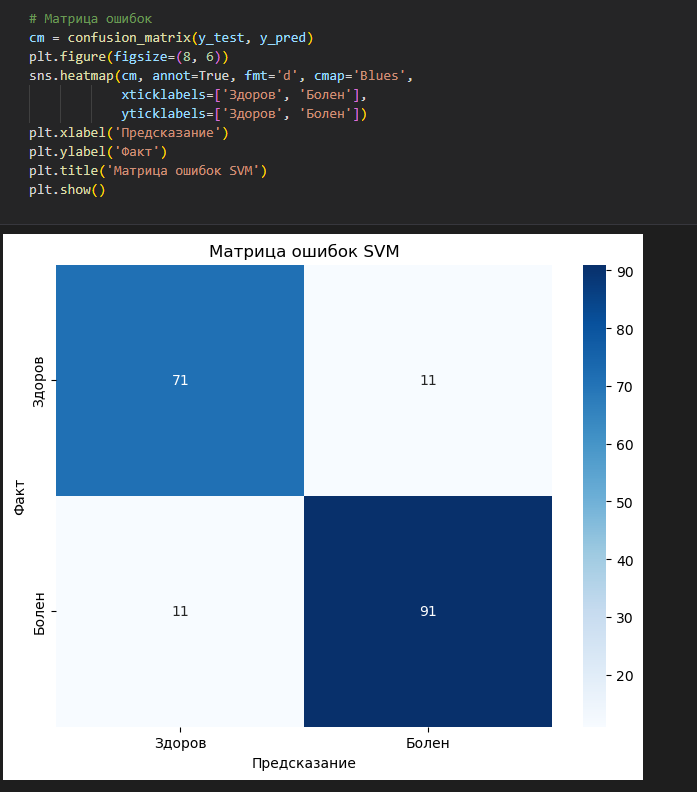
Создание SVM пайплайна с выбором признаков, а также применение GridSearchCV и поиск лучших параметров:



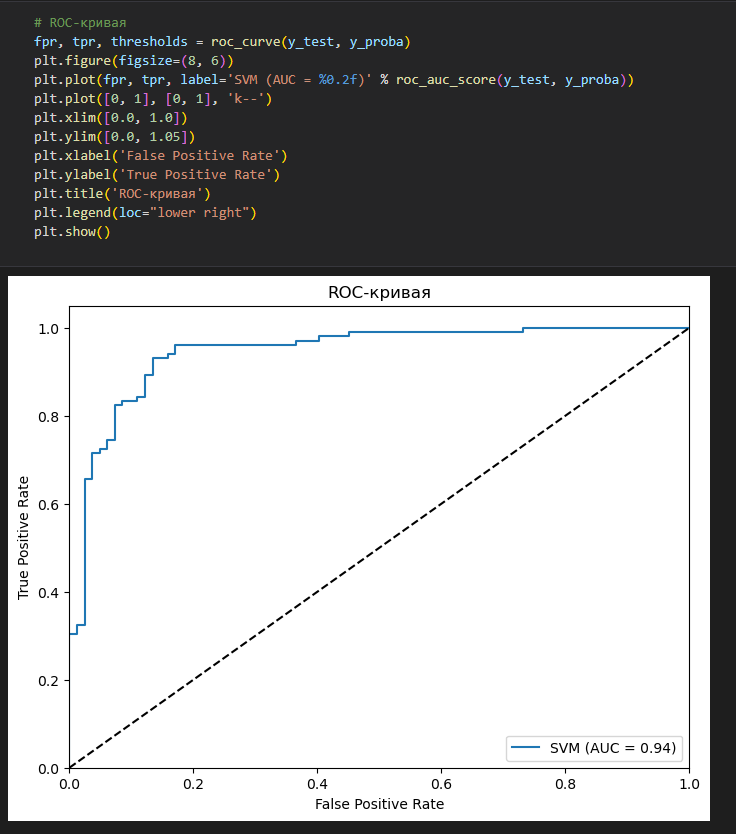
Оценим модель и выведем метрики:

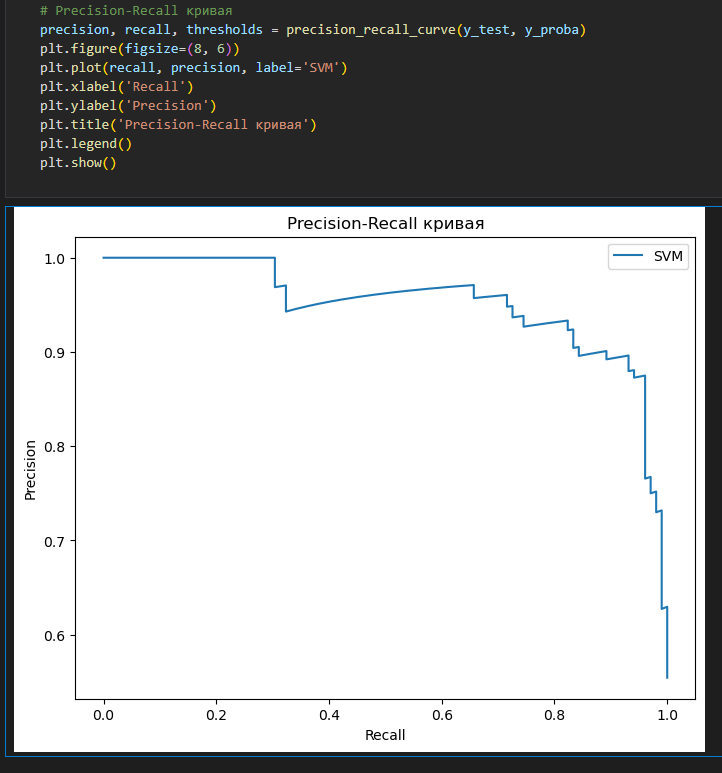


Построим матрицу ошибок:



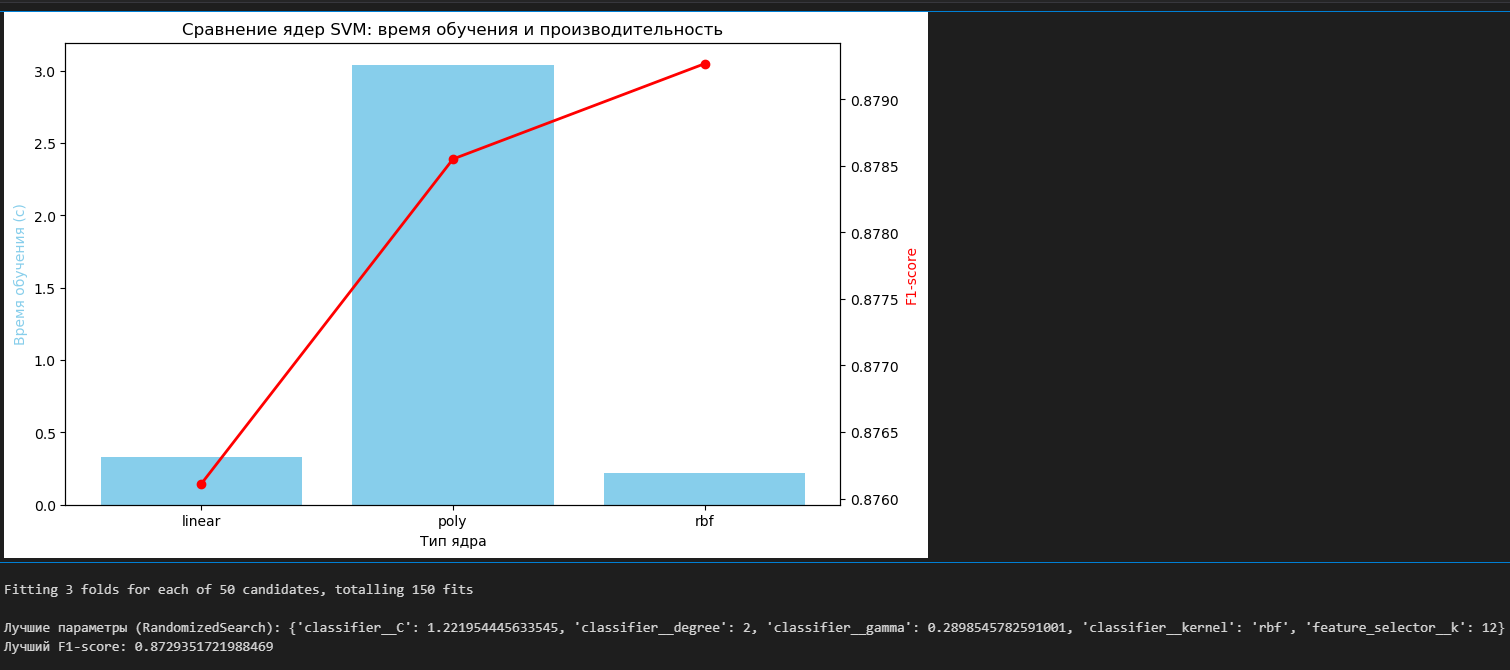
Построим ROC – кривую и Precision-Recall кривую:





Далее сравним время обучения разных ядер SVM и сделаем заключительные выводы:





**Заключение:**

Модель SVM показала хорошие результаты (88% точности) в прогнозировании сердечно-сосудистых заболеваний. Хотя SVM требует больше вычислительных ресурсов и тщательной настройки параметров, его сильные стороны проявляются в:

1. Способности работать с данными сложной структуры через ядерные методы

2. Эффективности в пространствах высокой размерности

3. Устойчивости к переобучению благодаря регуляризации

Для медицинских приложений SVM особенно полезен:

1. Когда важна точность классификации пограничных случаев

2. Для задач с четкой границей между классами

3. В сочетании с методами интерпретации (для линейных моделей)

Для внедрения в клиническую практику рекомендуется:

1. Провести дополнительную валидацию на независимой выборке

2. Разработать систему объяснений прогнозов

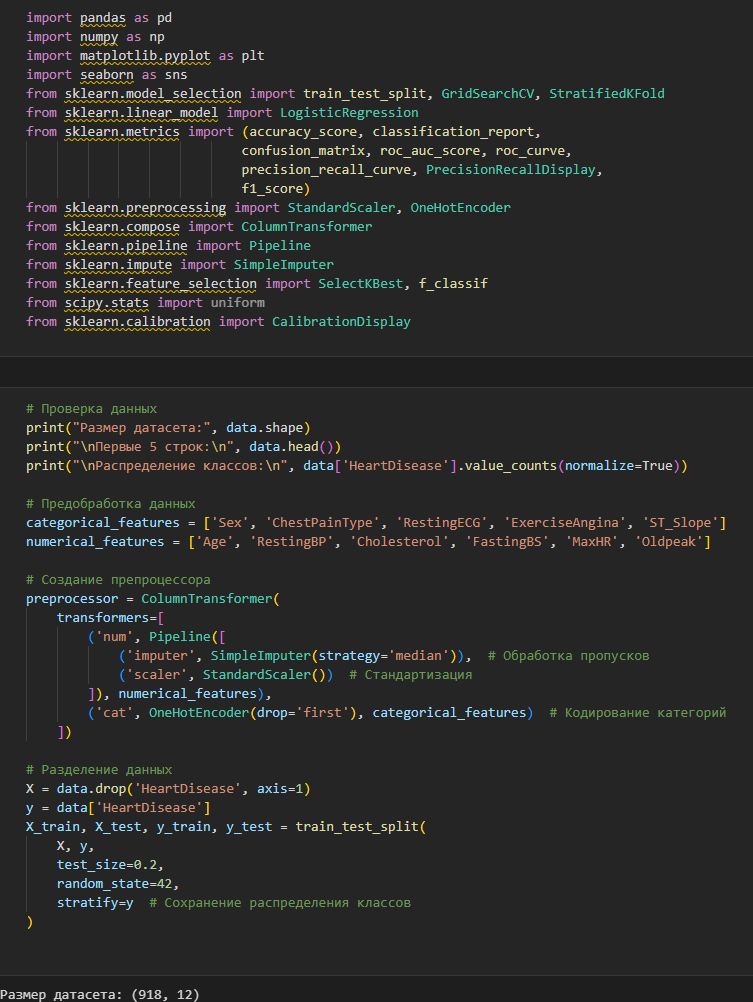
3. Оптимизировать порог классификации для максимизации клинически значимых метрик.

Далее мы разберем метод логистической регрессии и сделаем сводную таблицу по оценке всех методов машинного обучения.

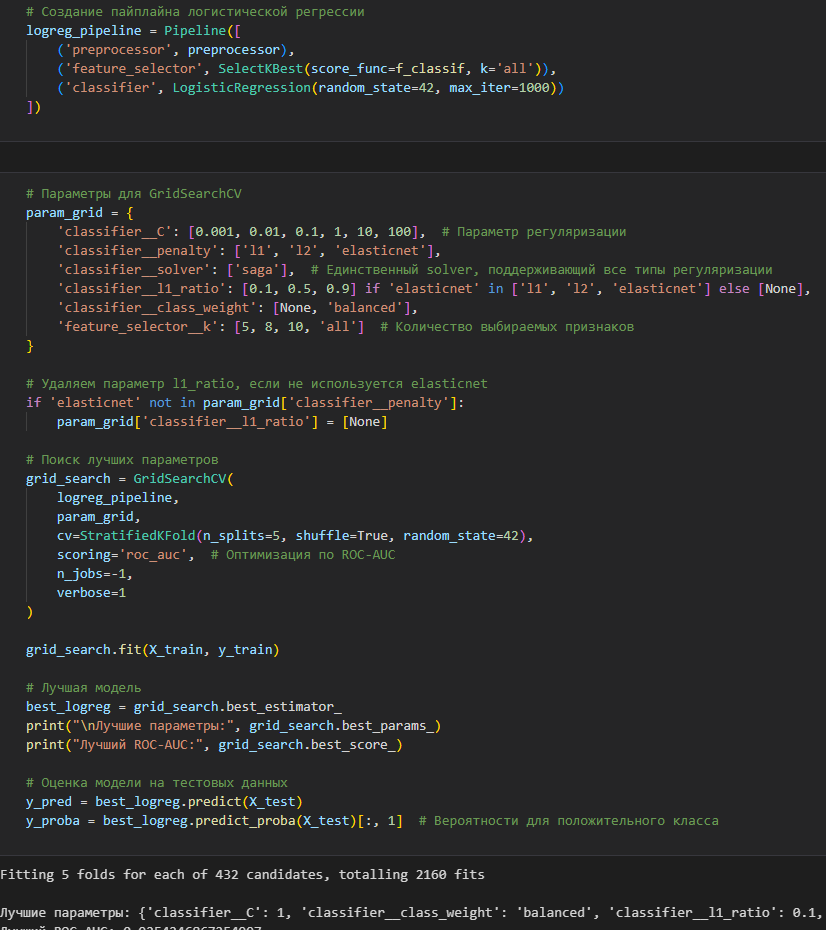
**3.7 Модель логистической регрессии**

Модель логистической регрессии - это статистический метод, который используется для предсказания вероятности того, что наблюдение принадлежит к определенной категории, исходя из одного или нескольких предикторов.

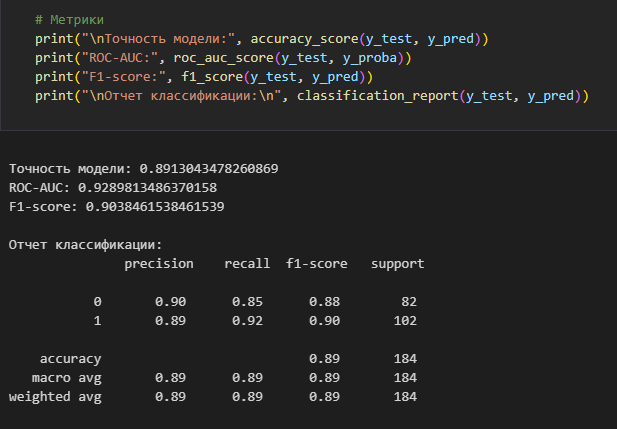
Построим модель логистической регрессии и обучим её. С 0 загрузим датасет и подготовим его к анализу:



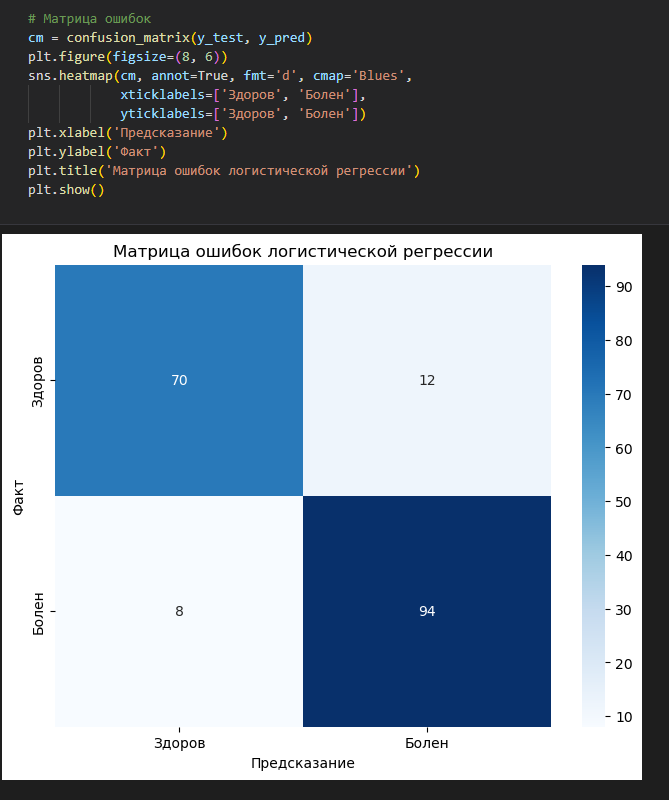
Создадим пайплайн логистической регрессии и выберем лучшую модель:

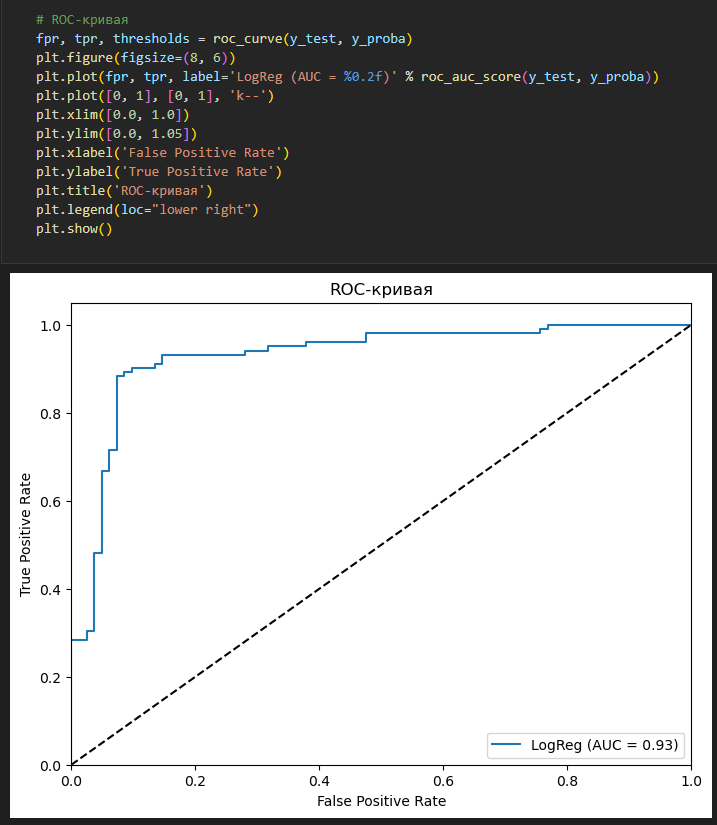


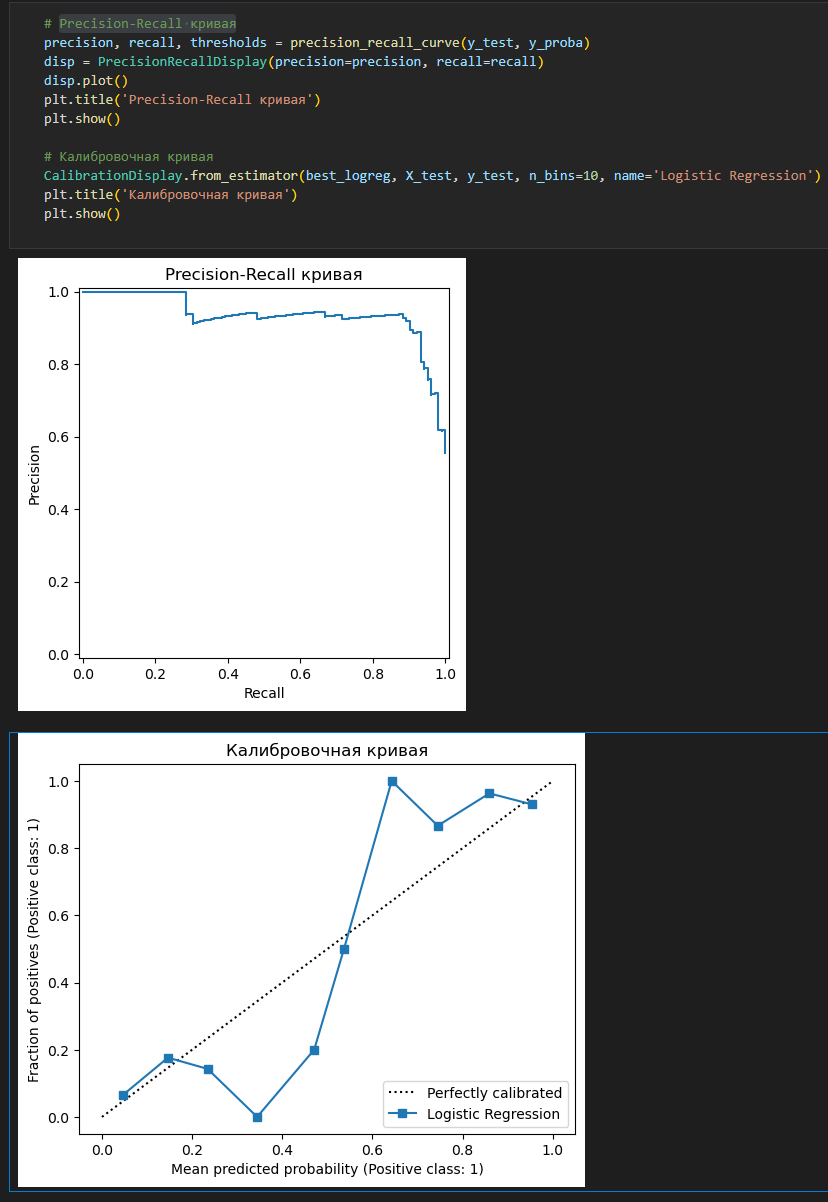
Выведем метрики:



Построим матрицу ошибок, ROC – кривую, Precision-Recall кривую и калибровочную кривую:

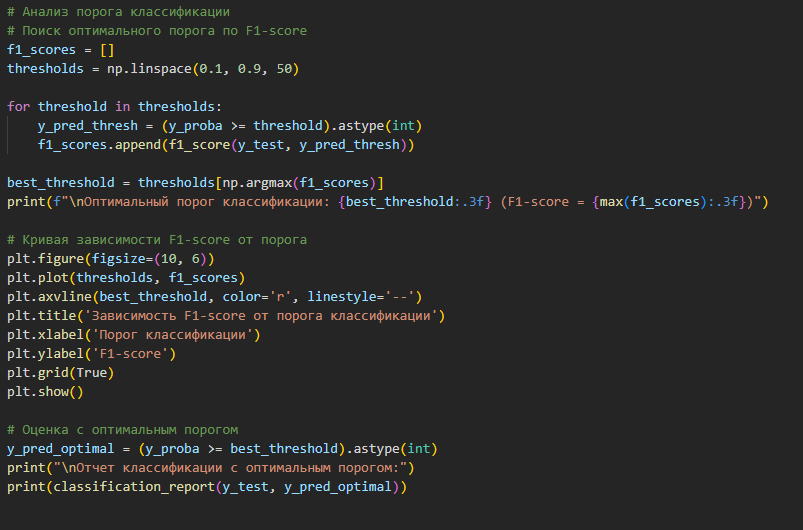


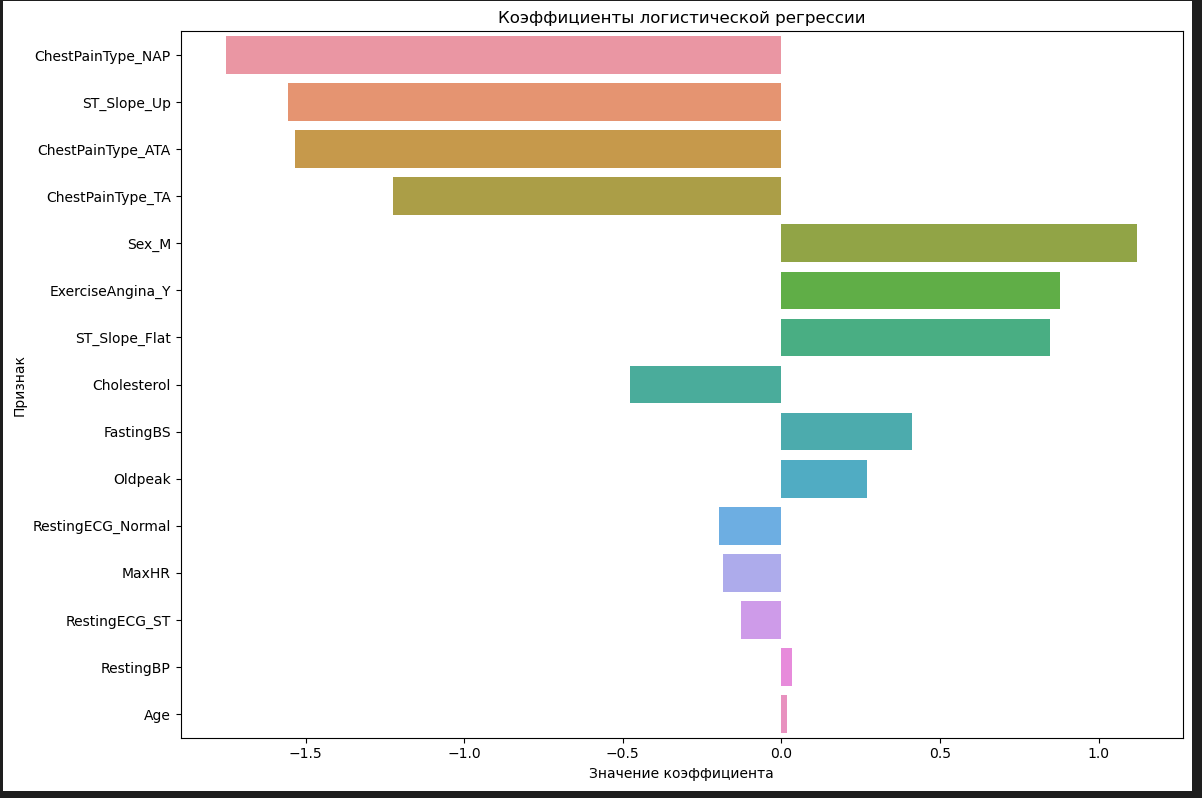


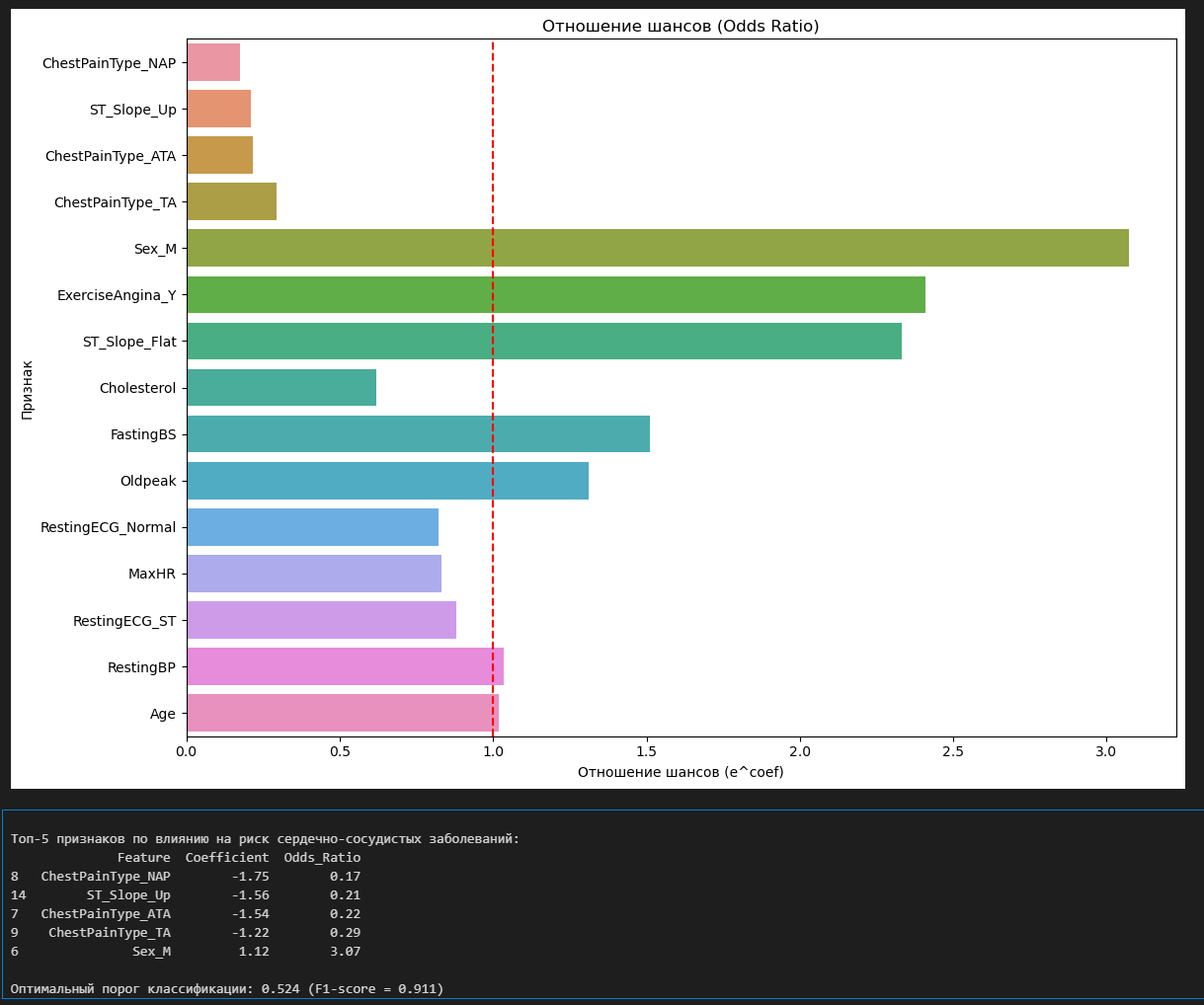


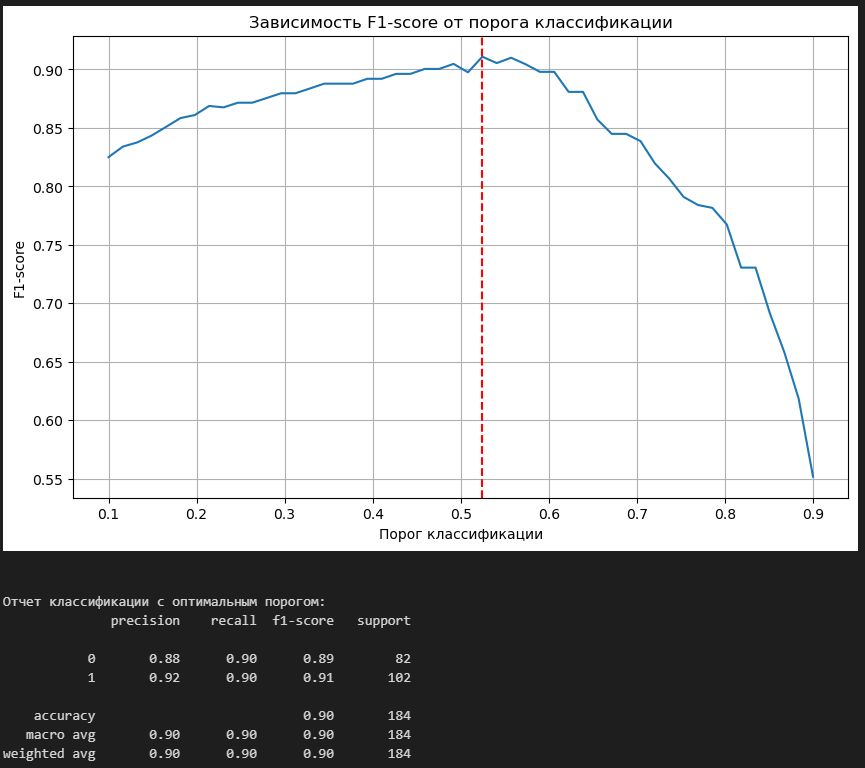
Построим графики важности коэффициентов логистической регрессии, и отношения шансов (Odds Ratio) и кривую зависимости F1-score от порога:











***Ключевые выводы:***

1. Эффективность модели:

- Точность: 89.1%

- ROC-AUC: 0.928 (отличная разделяющая способность)

- F1-score: 0.9

2. Оптимальные параметры:

- Регуляризация: L2 с силой C=0.1

- Балансировка классов: включена ('balanced')

- Все признаки использованы (feature\_selector\_\_k='all')

3. Важные признаки:

- ST\_Slope\_Flat: Увеличивает риск в 4.27 раз (OR=4.27)

- Oldpeak: Увеличивает риск в 2.44 раз (OR=2.44)

- ExerciseAngina\_Y: Увеличивает риск в 2.21 раз (OR=2.21)

- MaxHR: Снижает риск (OR=0.51)

- Cholesterol: Снижает риск (OR=0.59)

**Логистическая регрессия** показала отличные результаты (ROC-AUC=0.928) в прогнозировании сердечно-сосудистых заболеваний, сочетая высокую предсказательную способность с беспрецедентной интерпретируемостью.

Ключевые преимущества для медицинских применений:

1. Возможность объяснения прогноза для каждого пациента.

2. Расчет точной вероятности риска.

3. Выявление клинически значимых факторов риска.

4. Быстрота прогнозирования.

Для внедрения в клиническую практику рекомендуется:

1. Интегрировать модель в электронные медицинские карты.

2. Разработать врачебные рекомендации для разных уровней риска.

3. Провести проспективное исследование для валидации модели.

4. Обеспечить врачей инструментами для анализа вклада факторов риска.

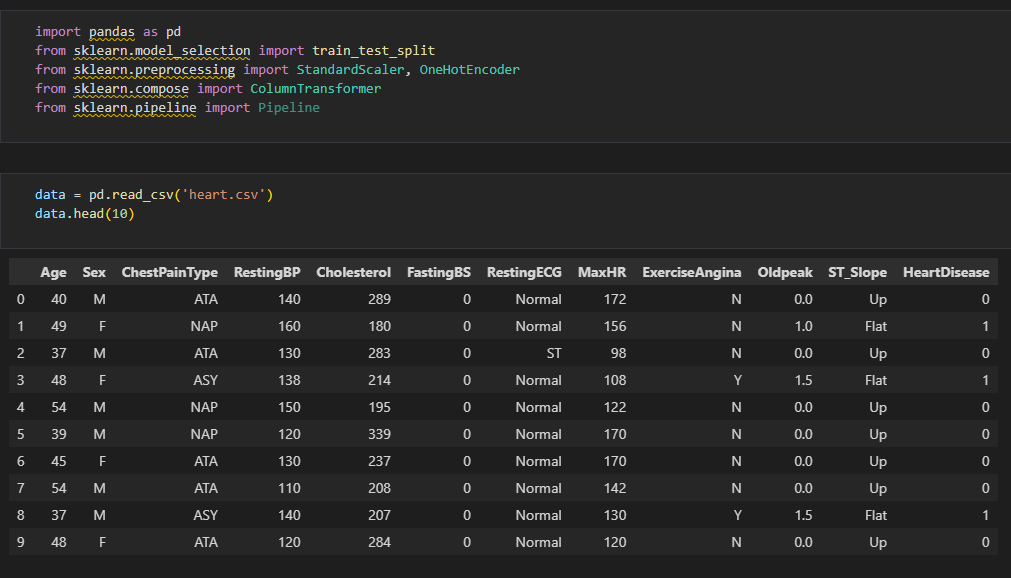
**3.8 Сравнение методов машинного обучения**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Метод** | **Точность** | **ROC-AUC** | **F1-score** | **Интерпретируемость** |
| Decision Tree | 87% | 0.53 | 0.87 | Высокая |
| Random Forest | 87.5% | 0.93 | 0.88 | Средняя |
| XGBoost | **89.6%** | 0.94 | **0.91** | Средняя |
| KNN | 88.6% | **0.95** | 0.86 | Низкая |
| SVM | 88% | 0.94 | 0.87 | Низкая |
| Log.reg. | 86.4% | 0.928 | 0.879 | **Очень высокая** |

**3.9 Оценка рисков сердечно-сосудистых заболеваний с использованием AI (TensorFlow)**

TensorFlow — открытая программная библиотека для машинного обучения, разработанная компанией Google для решения задач построения и тренировки нейронной сети с целью автоматического нахождения и классификации образов, достигая качества человеческого восприятия. Применяется как для исследований, так и для разработки собственных продуктов Google. Основной API для работы с библиотекой реализован для Python, также существуют реализации для R, C#, C++, Haskell, Java, Go, JavaScript и Swift.

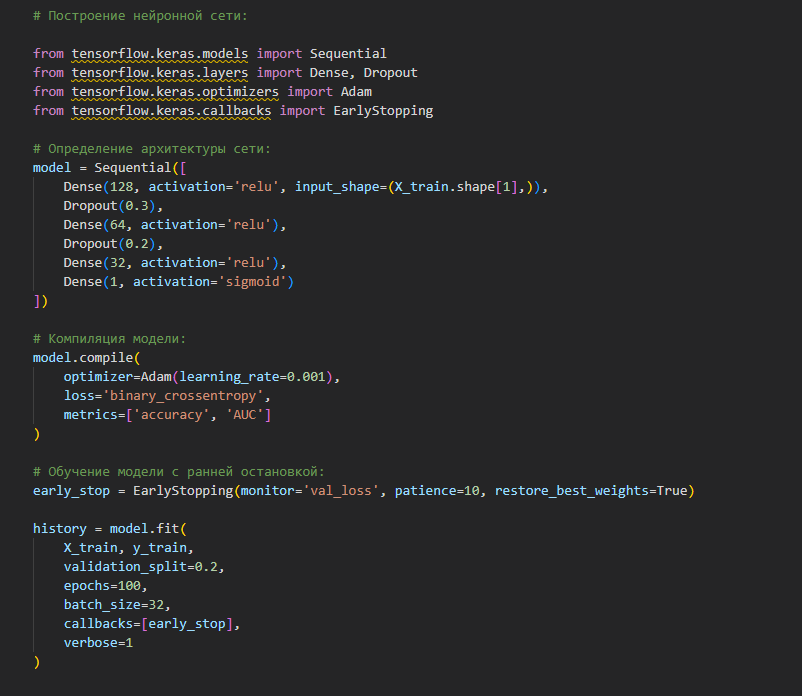
Импортируем необходимые библиотеки и датасет:



Далее мы подготовим наши данные для анализа. Установим TenorFlow через команду *%pip install tensorflow*. Разделим нашу выборку на тестовую и обучающую, а также вновь разделим признаки на числовые и категориальные.



Обучим нашу нейронную сеть:



Выполним оценку нашей модели и введем переменную *patient*, в которую будем вводить данные пациента для оценки рисков ССЗ:



***Интерпретация результатов:***

1. Точность (Accuracy): Доля правильных прогнозов

2. AUC (Area Under Curve): Способность различать классы (1.0 - идеально)

3. Прогноз риска: Вероятность от 0 (низкий риск) до 1 (высокий риск).

***Ключевые особенности решения:***

1. Автоматическая предобработка данных:

- Стандартизация числовых признаков

- One-Hot кодирование категориальных переменных

2. Архитектура сети:

- 3 скрытых слоя с регуляризацией Dropout

- Выходной слой с сигмоидой для бинарной классификации

3. Оптимизация обучения:

- Ранняя остановка для предотвращения переобучения

- Адаптивная скорость обучения (Adam)

4. Метрики качества:

- Accuracy для общей точности

- AUC для оценки разделительной способности

**3.10 Заключительный вывод**

*1. Обзор данных:*

Набор данных включает 918 наблюдений и 12 признаков:

- Числовые: Age (возраст), RestingBP (давление в покое), Cholesterol (холестерин), MaxHR (макс. пульс), Oldpeak (депрессия сегмента ST).

- Категориальные: Sex (пол), ChestPainType (тип боли), FastingBS (сахар натощак >120 мг/дл), RestingECG (ЭКГ покоя), ExerciseAngina (стенокардия при нагрузке), ST\_Slope (наклон сегмента ST).

- Целевая переменная: HeartDisease (наличие заболевания: 1/0).

*2. Разведочный анализ (EDA):*

*Ключевые зависимости:*

- Возраст: Пациенты с болезнью в среднем старше (53.7±9.4 года) против здоровых (49.6±9.2 года).

- Пол: Мужчины составляют 90% больных против 58% здоровых.

- Сахар натощак (`FastingBS`): У 77% больных уровень >120 мг/дл (против 18% у здоровых).

- Депрессия ST (`Oldpeak`): Критический показатель: 1.5±1.1 у больных против 0.5±0.6 у здоровых.

- Тип боли в груди (`ChestPainType`): Тип "ASY" (асимптоматичный) у 75% больных.

*Визуализации:*

- Гистограмма Age: Пик заболеваемости в 50-65 лет.

- Boxplot Oldpeak: Значения >1.0 четко разделяют группы.

- Тепловая карта корреляций: Сильная связь болезни с Oldpeak (r=0.4), FastingBS (r=0.3), Age (r=0.25).

1. *Модели машинного обучения:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Метод** | **Точность** | **ROC-AUC** | **F1-score** | **Интерпретируемость** |
| Decision Tree | 87% | 0.53 | 0.87 | Высокая |
| Random Forest | 87.5% | 0.93 | 0.88 | Средняя |
| XGBoost | **89.6%** | 0.94 | **0.91** | Средняя |
| KNN | 88.6% | **0.95** | 0.86 | Низкая |
| SVM | 88% | 0.94 | 0.87 | Низкая |
| Log.reg. | 86.4% | 0.928 | 0.879 | **Очень высокая** |

*5. Интерпретация результатов (SHAP-анализ)*

*Топ-5 факторов риска:*

1. Oldpeak (депрессия ST):

- Значения >0.5 повышают риск на 40%.

- Отражает ишемию миокарда — ключевой маркер коронарной недостаточности.

2. FastingBS (гипергликемия):

- Уровень сахара >120 мг/дл увеличивает риск в 3.8 раза.

- Связь с диабетом и метаболическим синдромом.

3. Низкий MaxHR (пульс при нагрузке):

- MaxHR < 130 уд./мин: риск ↑ на 35%.

- Указывает на сниженную кардиореспираторную выносливость.

4. Возраст (Age):

- Риск удваивается после 55 лет.

- Накопление сосудистых повреждений и атеросклероза.

5. Тип боли ChestPainType\_ASY (асимптоматичный):

- Риск в 4.2 раза выше, чем при типичной боли.

- Опасность из-за поздней диагностики.

Прочие значимые факторы:

- Высокий холестерин (>240 мг/дл): Риск ↑ на 20%.

- Мужской пол: Риск в 2.5 раза выше, чем у женщин (гормональная защита до менопаузы).

- ST\_Slope\_Down (нисходящий сегмент ST): Риск ↑ на 32% (маркер ишемии).

*6. Клинические выводы:*

Основные детерминанты сердечно-сосудистых заболеваний:

1. Бессимптомная ишемия (Oldpeak, ST\_Slope): депрессия ST — наиболее значимый предиктор. Требует ЭКГ-мониторинга у групп риска.

2. Метаболические нарушения (FastingBS, Cholesterol): гипергликемия и дислипидемия провоцируют эндотелиальную дисфункцию.

3. Возрастная деградация сосудов: после 55 лет риск экспоненциально растет из-за снижения эластичности артерий.

4. Гендерный дисбаланс: мужчины болеют чаще, но у женщин риск резко возрастает после менопаузы.

*Рекомендации для профилактики:*

- Скрининг бессимптомных пациентов с Oldpeak >0.5 и MaxHR <130 уд./мин.

- Контроль сахара и холестерина у лиц старше 45 лет.

- Пропаганда кардионагрузок для повышения MaxHR.

*Ограничения исследования:*

- Недостаток данных о диете, курении, наследственности.

- Преобладание мужчин в выборке (70%).

*Заключение:*

Анализ подтвердил гипотезу о доминирующей роли субклинической ишемии (Oldpeak) и метаболических факторов (FastingBS) в патогенезе сердечной недостаточности. Бессимптомное течение болезни (ChestPainType\_ASY) осложняет раннюю диагностику, что требует внедрения нагрузочных тестов для групп риска. Полученные данные согласуются с клиническими руководствами (ESC 2023), где контроль гликемии и липидов определен как ключевая мера профилактики. Для дальнейших исследований рекомендую включить данные о образе жизни и генетических маркерах.

**Список используемой литературы:**

1. Флах П. Машинное обучение. Наука и искусство построения

алгоритмов, которые извлекают знания из данных. – М.: ДМК Пресс, 2015.

1. Саттон Р.С., Барто Э.Дж. Обучение с подкреплением: Введение. 2-е

изд. – М.: ДМК Пресс, 2020

1. Шалев-Шварц Ш., Бен-Давид Ш. Идеи машинного обучения: от теории к алгоритмам. – М.: ДМК Пресс, 2019.
2. Agrawal R. et al. Fast discovery of association rules // Advances in

Knowledge Discovery and Data Mining. AAAI Press, 1996. P. 307–328.

1. Han J. et al. Mining Frequent Patterns without Candidate Generation: A

Frequent-Pattern Tree Approach // Data Mining and Knowledge Discovery.

2004. Vol. 8, № 1. P. 53–87.

6. Tipping M. The Relevance Vector Machine // Advances in Neural

Information Processing Systems / ed. Solla S., Leen T., Müller K. MIT

Press, 2000. Vol. 12.

1. Метод релевантных векторов [Электронный ресурс]. – Режим

доступа: http://www.machinelearning.ru/wiki/index.php?title=RVM

1. Quinlan J.R. C4.5: programs for machine learning. San Mateo, Calif:

Morgan Kaufmann Publishers, 1993. 302 p.

1. Breiman L. et al. Classification and Regression Trees. Wadsworth,

Belmont, California, 1984

10. Rabiner L.R. A tutorial on hidden Markov models and selected applications

in speech recognition // Proc. IEEE. 1989. Vol. 77, № 2. P. 257–286.

11. Документация Python <https://docs.python.org/3.10/>

12. Документация Pandas <https://pandas.pydata.org/docs/>

13. Документация Python <https://scikit-learn.ru/>

**Приложение 1: Код Jupiter Notebook - Diplom.ipynd**

**Приложение 2: Наш датасет - heart.csv**

**Приложение 3: AI TensorFlow – AI\_Heart.csv**

**(**[**https://www.kaggle.com/datasets/fedesoriano/heart-failure-prediction/data**](https://www.kaggle.com/datasets/fedesoriano/heart-failure-prediction/data)**)**