ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧЕРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ УФИМСКИЙ УНИВЕРСИТЕТ НАУКИ И ТЕХНОЛОГИЙ

**ОТЧЕТ**

по итоговому проекту курса «Технологии обработки количественных данных»

**«Прогнозирование возникновения сердечного приступа у пациента»**

**Уфа 2023 г.**

**Содержание**

[Содержательная постановка задачи 3](#_Toc172417435)

[Описание входных и выходных данных 5](#_Toc172417436)

[Сравнительный анализ методов и алгоритмов 6](#_Toc172417437)

[Формальная постановка задачи 8](#_Toc172417438)

[Предобработка данных 10](#_Toc172417439)

[Логистическая регрессия 15](#_Toc172417440)

[Метод k-ближайших соседей 22](#_Toc172417441)

[Дерево решений 27](#_Toc172417442)

[Анализ методов и алгоритмов обучения 31](#_Toc172417443)

[Заключение 35](#_Toc172417444)

# Содержательная постановка задачи

Содержательная постановка задачи проектной работы заключается в разработке эффективной модели прогнозирования возникновения сердечного приступа у пациента на основе анализа медицинских данных из датасета, доступного на платформе Kaggle («Heart Attack Analysis & Prediction Dataset»).

Для достижения данной цели требуется выполнить следующие задачи:

1. Сбор и предобработка данных.

Извлечение данных из исходного файла и их предобработка, включая заполнение пропущенных значений, нормализацию числовых признаков и кодирование категориальных признаков. Проведение исследовательского анализа данных для выявления закономерностей и корреляций между признаками, а также подготовка данных к обучению моделей.

1. Анализ существующих методов и решений.

Изучение существующих подходов и опыта в сфере реализации задач прогнозирования в медицине, анализ преимуществ и недостатков используемых подходов. Выбор нескольких методов машинного обучения для дальнейшего исследования в сравнительном анализе моделей для прогнозирования риска сердечного приступа.

1. Разработка прогностических моделей.

Обучить выбранные модели на предобработанных данных, используя разделение данных на обучающую и тестовую выборки.

1. Оценка моделей.

Провести оценку эффективности моделей с помощью оценки точности, полноты, F1-меры, кросс-валидации, матрицы путаницы, а также построение ROC-кривых и вычисление площади под ними (AUC-ROC).

1. Сравнительный анализ и интерпретация результатов.

Провести сравнительный анализ результатов работы различных моделей и выбрать наиболее эффективную модель для прогнозирования риска сердечного приступа. Проанализировать важность признаков, полученных от выбранной модели, для лучшего понимания факторов, влияющих на риск сердечного приступа.

Результаты работы будут способствовать более точному определению риска сердечного приступа, что имеет важное значение для здравоохранения и предотвращения серьезных заболеваний у пациентов.

# Описание входных и выходных данных

Был использован набор данных для анализа и прогнозирования риска сердечного приступа с портала Kaggle: <https://www.kaggle.com/datasets/rashikrahmanpritom/heart-attack-analysis-prediction-dataset>.

**Входные данные:**

age – возраст пациента,

sex – пол пациента,

cp – тип боли в груди,

trtbps - артериальное давление в состоянии покоя (в мм рт. ст.),

chol - холесторол в мг/дл, полученный с помощью датчика ИМТ,

fbs - уровень сахара в крови натощак > 120 мг/дл (1 = верно; 0 = неверно),

restecg - результаты электрокардиографии в покое,

thalachh - максимальная частота сердечных сокращений,

exng - стенокардия, вызванная физической нагрузкой (1 = да; 0 = нет),

oldpeak -снижение ST, вызванное нагрузкой, относительно покоя,

slp – slope, наклон (ST/частота сердечных сокращений) был предложен в качестве более точного ЭКГ-критерия для диагностики значимой ишемической болезни сердца (ИБС),

caa - number of major vessels (0-3) - количество магистральных сосудов, окрашенных при рентгеноскопии,

thall – коэффициент талассемии.

**Выходные данные:**

output – риск сердечного приступа.

# Сравнительный анализ методов и алгоритмов

Проведен сравнительный анализ нескольких методов машинного обучения в контексте задачи прогнозирования сердечного приступа на небольшом наборе данных, состоящем из 300 строк и 14 параметров. Были рассмотрены следующие методы:

* Дерево решений основано на разбиении набора данных на более мелкие подгруппы на основе признаков и последовательном принятии решений. Дерево решений просто в понимании и интерпретации, а также способно работать как с категориальными, так и с числовыми данными. Недостаток заключается в том, что деревья решений могут быть склонны к переобучению, особенно на более крупных наборах данных, и не всегда находят оптимальное решение;
* Метод k-ближайших соседей определяет класс объекта на основе классов его k-ближайших соседей. Среди преимуществ можно выделить, что k-NN прост в реализации и понимании и может работать с разнообразными типами данных. Однако он чувствителен к шуму и выбросам и может быть неэффективен на больших наборах данных из-за вычислительных затрат;
* Логистическая регрессия используется для моделирования вероятности бинарного класса. Этот метод эффективен для бинарной классификации, устойчив к выбросам и предоставляет вероятности классов, но может не справляться с сложными нелинейными зависимостями;
* Метод опорных векторов среди рассмотренных методов выделяется своей способностью хорошо разделять классы в пространствах большой размерности. Стоит отметить, что данный метод требует тщательной настройки гиперпараметров, таких как ядро и параметр регуляризации C. Это может потребовать значительных вычислительных ресурсов и экспертных знаний. Кроме того, метод опорных векторов может быть сложным для интерпретации, так как оптимальная разделяющая гиперплоскость может быть сложной визуально;
* Случайный лес - алгоритм ансамбля, который часто считается устойчивым к переобучению и способным обрабатывать большие наборы данных, но также требует настройки параметров, таких как количество деревьев в ансамбле и глубина каждого дерева. Кроме того, он может не обеспечивать легкую интерпретируемость результатов, так как оценки важности признаков в случайных лесах могут быть неточными;
* Градиентный бустинг может обеспечивать высокую точность классификации и регрессии, что делает его мощным инструментом в медицинских прогнозных задачах. Однако этот метод может потребовать значительных вычислительных ресурсов и времени для обучения, особенно если требуется настройка большого числа деревьев и других параметров. Также, как и в случае случайного леса, интерпретация результатов может быть вызовом из-за сложности модели;
* Нейронные сети могут обеспечить высокую точность в задачах классификации и регрессии, но в данном контексте, ограниченном размером набора данных (300 строк), могут столкнуться с проблемой переобучения, особенно если сеть имеет множество параметров. Нейросети требуют значительных вычислительных ресурсов и времени для обучения. Интерпретация результатов также может быть сложной, особенно в случае глубоких сверточных или рекуррентных моделей.

В нашем случае из-за ограниченного размера набора данных и приоритета интерпретируемости модели, использование методов, таких как логистическая регрессия, дерево решений и метод k-ближайших соседей являются более предпочтительным.

# Формальная постановка задачи

На рисунке 1 представлена контекстная диаграмма методологии IDEF0, отражающая поставленную задачу, цель которой – выявить вероятность возникновения сердечного приступа у пациента.

Для функциональной модели IDEF0 0-го уровня детализации входными данными будут служить данные о пациенте. В качестве управляющих параметров будут использоваться требования к данным, алгоритм предобработки данных, метод логистической регрессии, метод k-NN и метод дерева решений.  В качестве механизмов выступают разработанные программные решения.

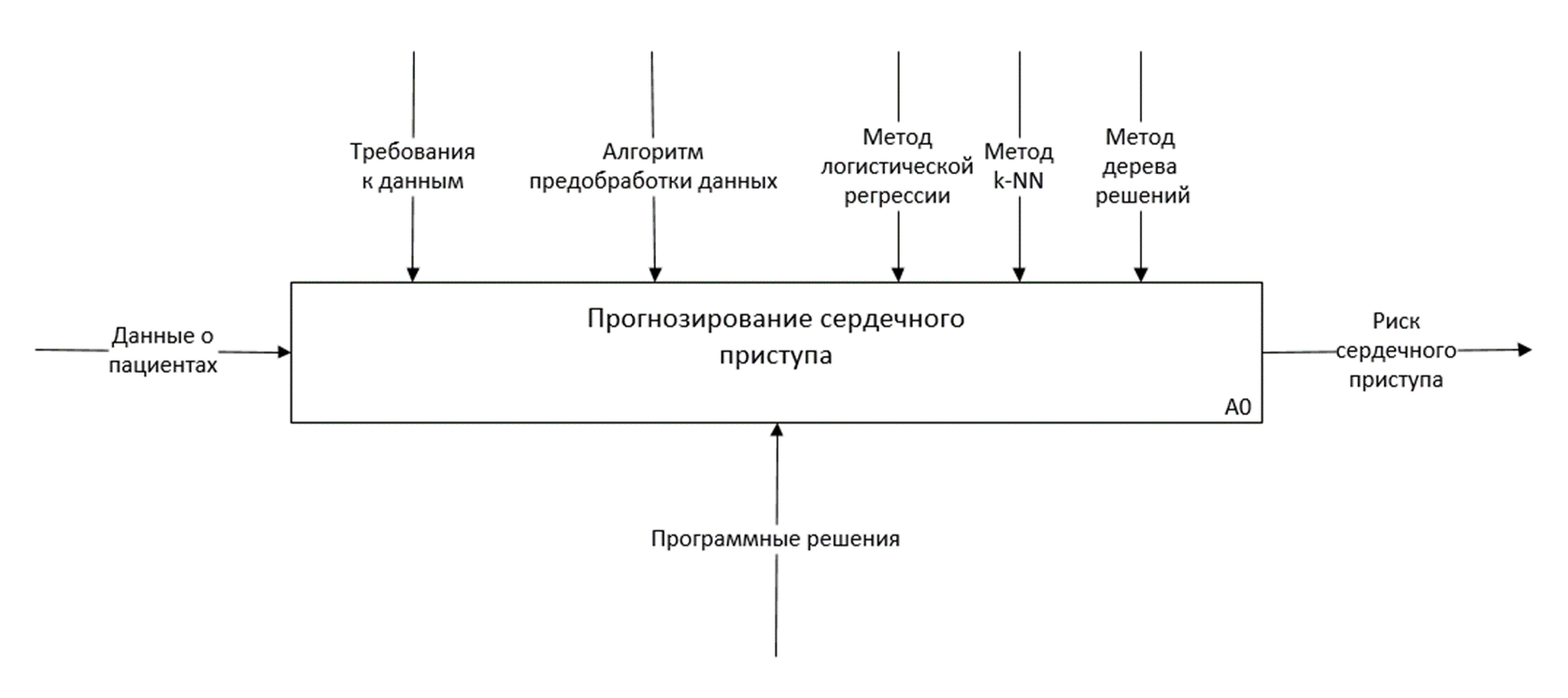


Рисунок 1 - Формальная постановка задачи прогнозирования сердечного приступа

Рассмотрим более подробно процесс прогнозирования сердечного приступа.  Декомпозиция задачи представлена на рисунке 2. Функциональная модель состоит из двух блоков: предобработка входных данных и обучение прогностических моделей. Входными параметрами второго блока являются обработанные входные данные.

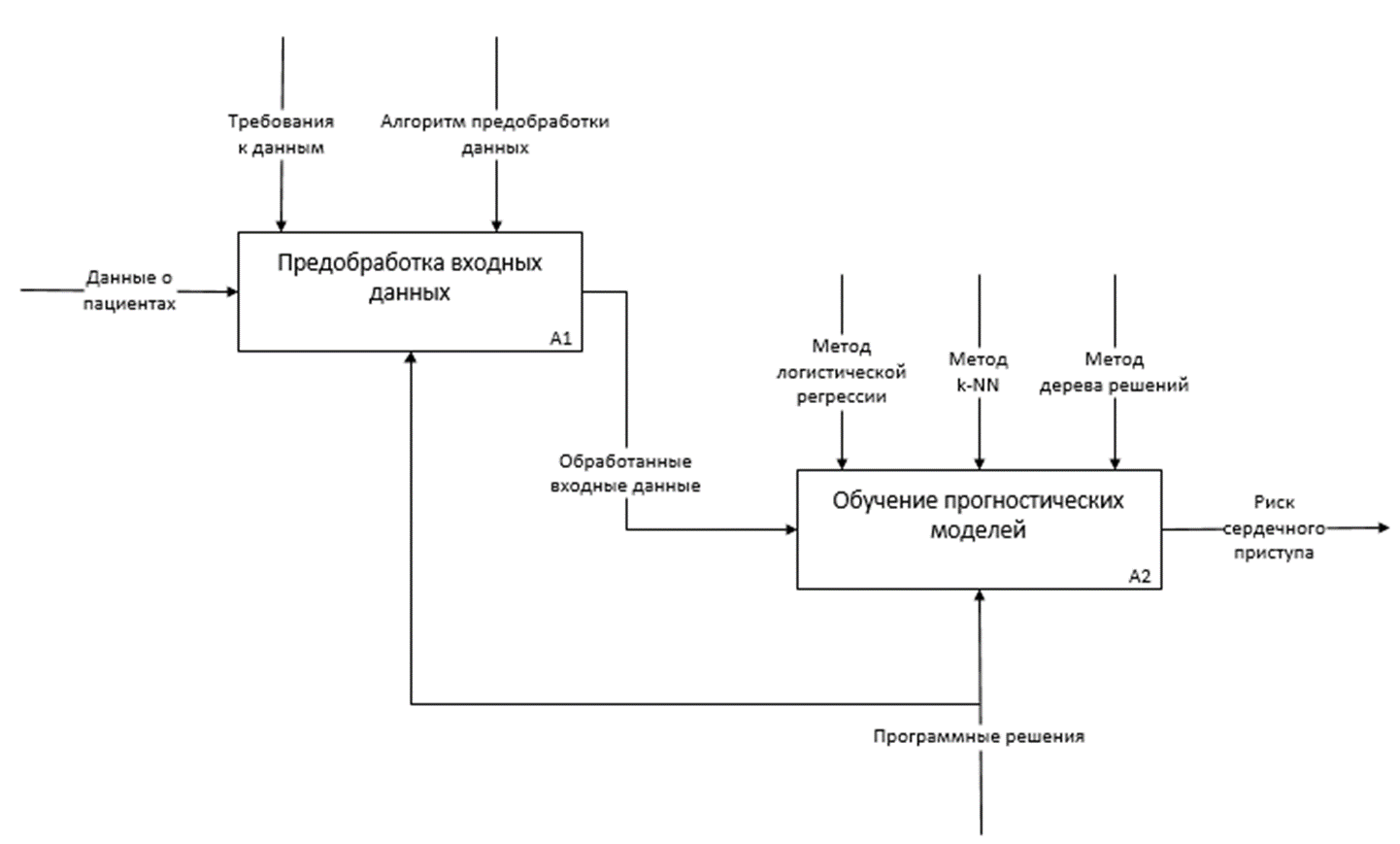


Рисунок 2 - Декомпозиция задачи

# Предобработка данных

После загрузки файла с датасетом heart.csv на Google Colab, используется библиотека Pandas для его анализа. Ниже представлен листинг кода с загрузкой исходного набора данных, выводом его первых строк и размера, а также информации о его структуре, включая типы данных и количество ненулевых значений, и основных статистических характеристик числовых столбцов (Рисунок 3-Рисунок 6).

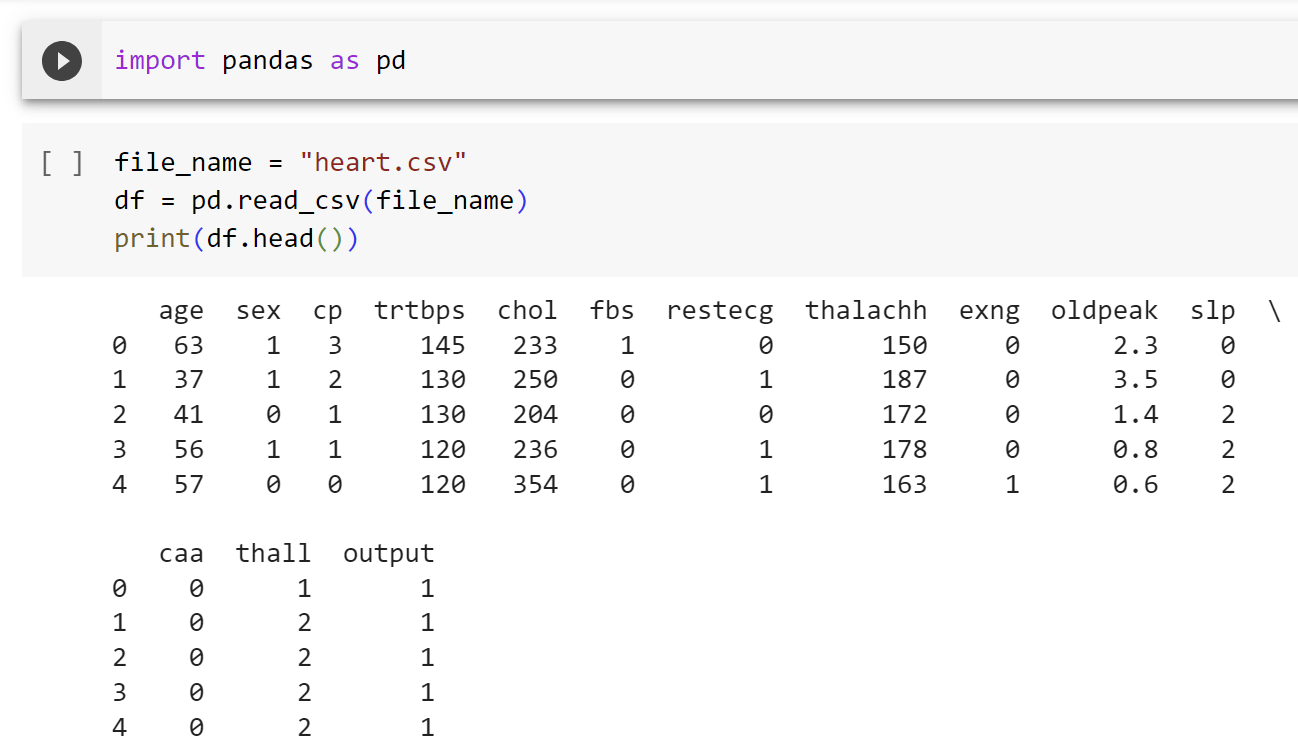


Рисунок 3 - Импорт датасета и вывод его первых строк

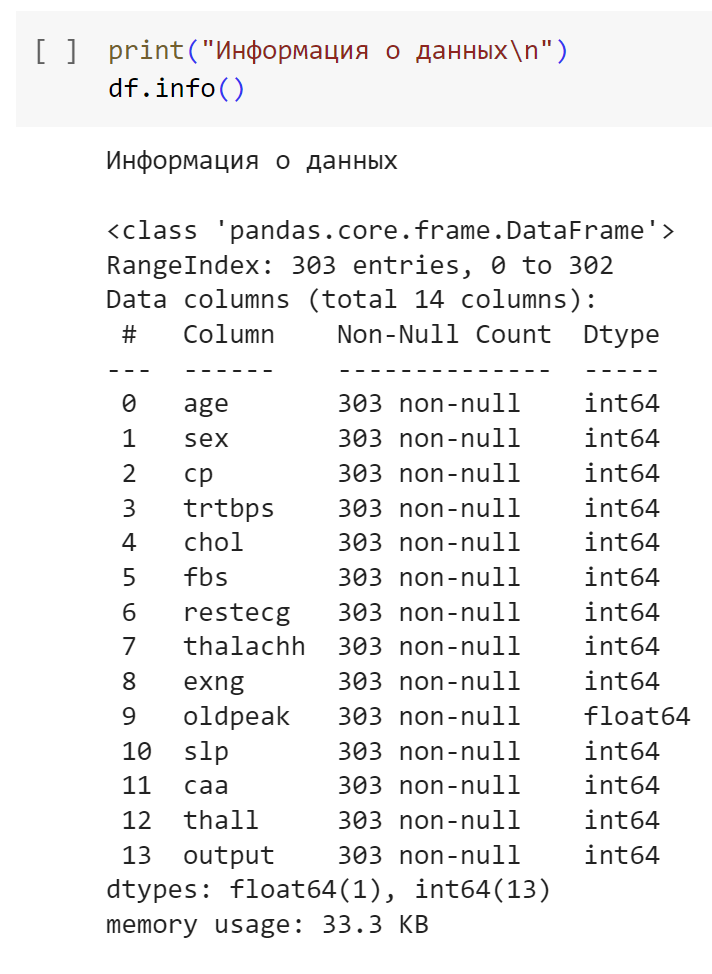


Рисунок 4 - Информация о структуре данных

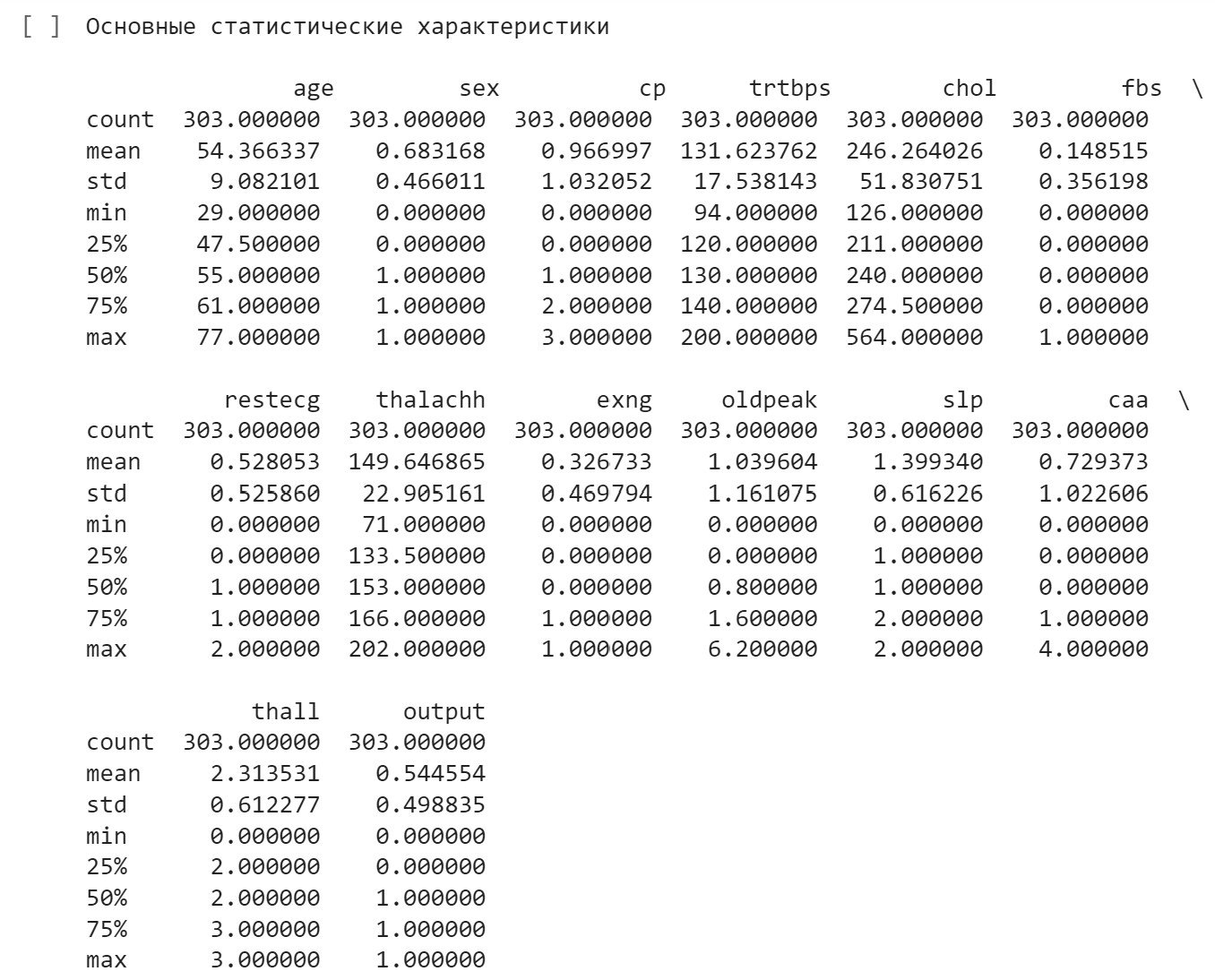


Рисунок 5 - Статистические характеристики числовых столбцов датасета

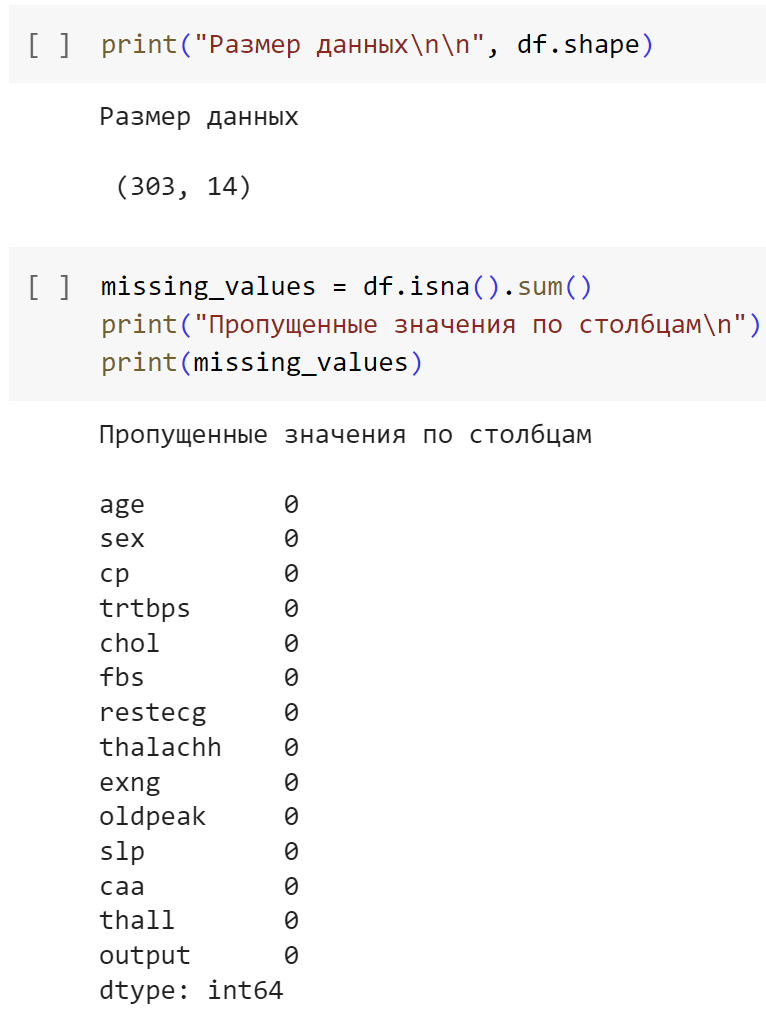


Рисунок 6 - Информация о размере данных и наличии пропусков

Выведены статистические данные для каждого столбца. С целью визуализации были построены столбчатые диаграммы распределения значений для числовых столбцов.

Нами было настроено количество столбцов гистограммы в зависимости от типа данных или характеристик для каждого конкретного параметра (Рисунок 7).

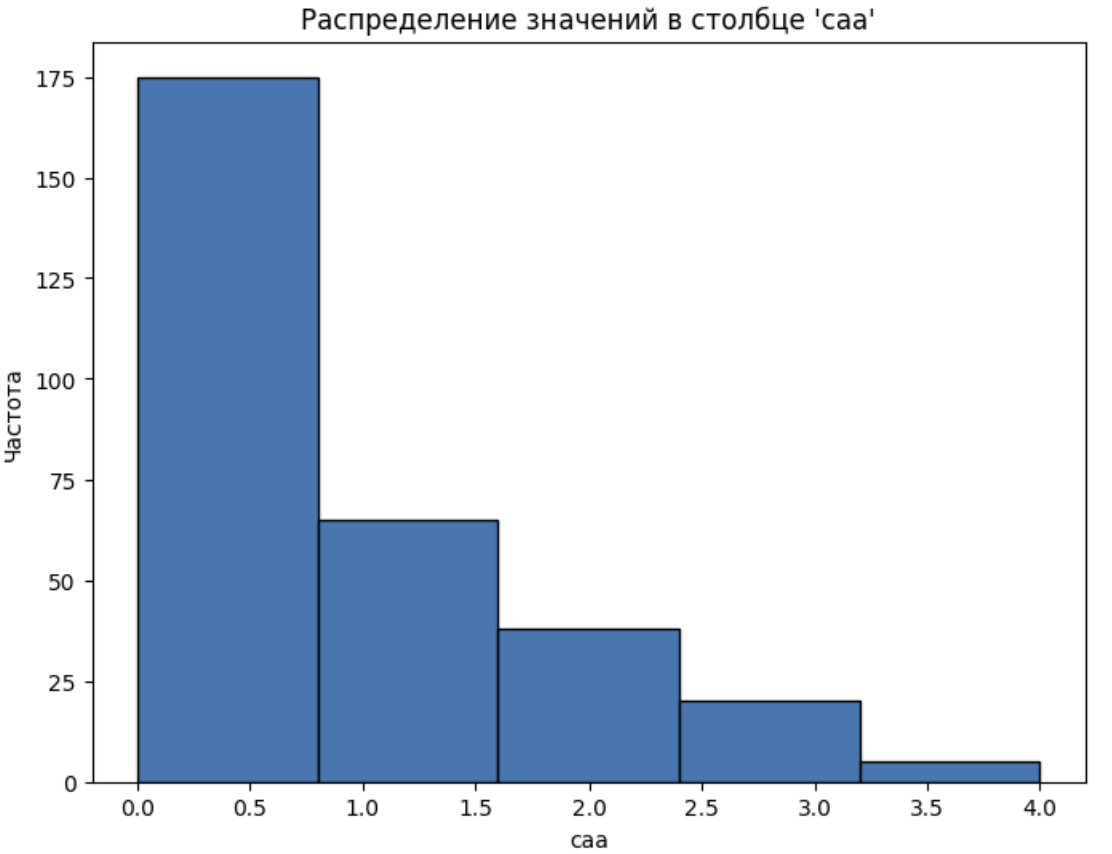
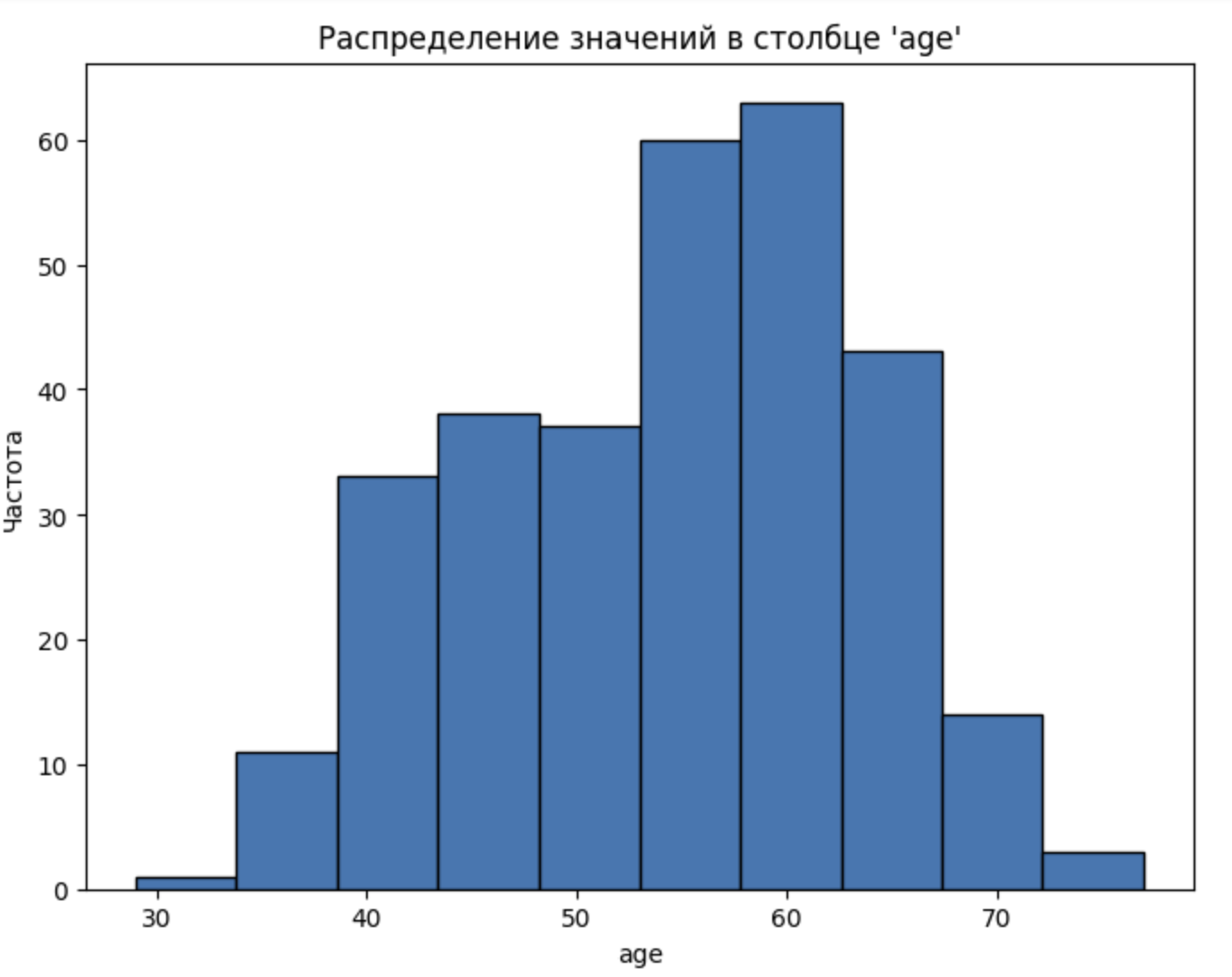


Рисунок 7 - Пример статистических данных столбцов age и caa

Для создания тепловой карты корреляции использовалась библиотека seaborn. Матрица корреляции представлена на Рисунке 8.

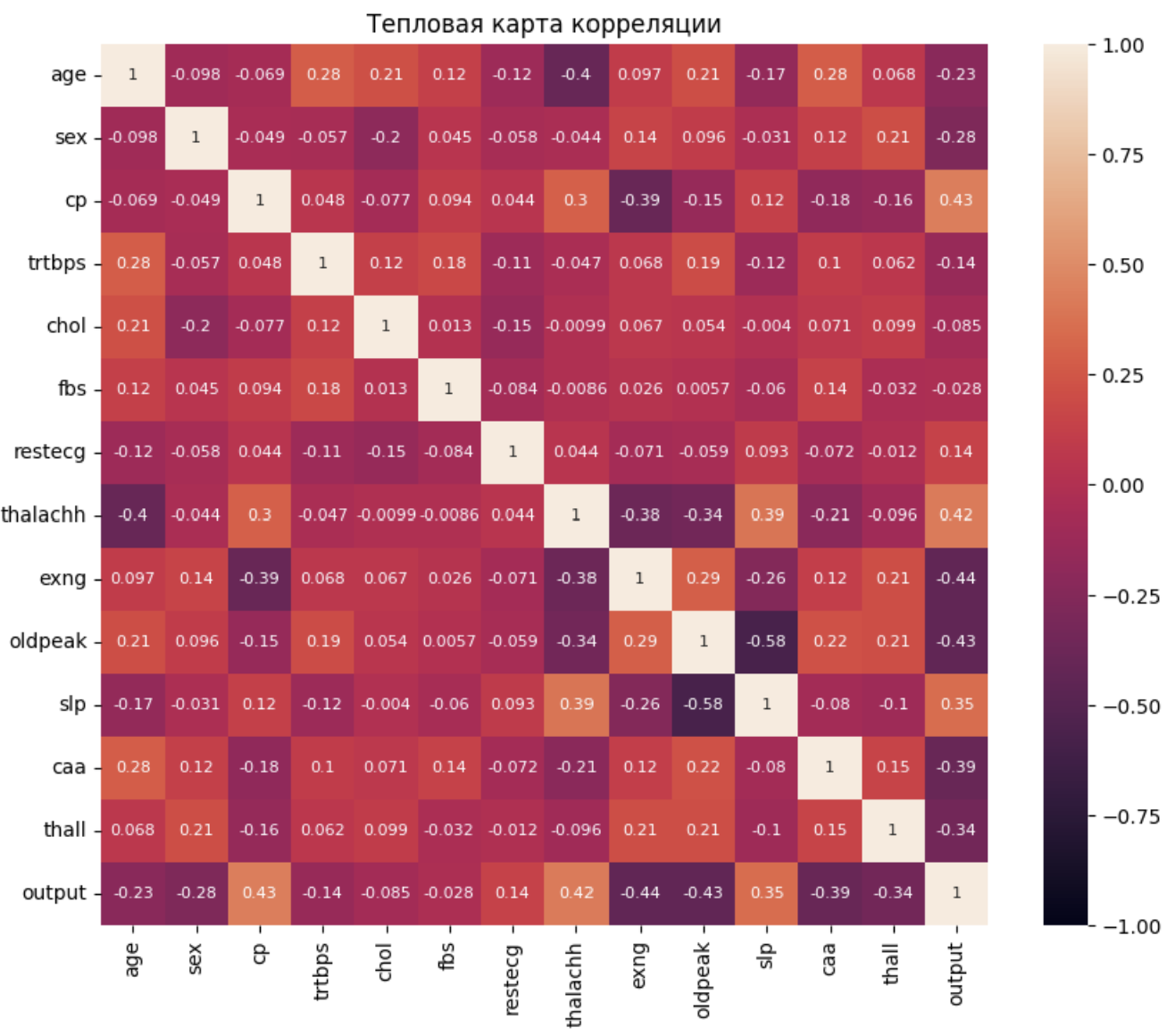


Рисунок 8 - Тепловая карта матрицы корреляции

Корреляционный анализ показал, что параметры chol и fbs имеют малую взаимосвязь с выявлением риска сердечного приступа, поэтому эти столбцы были удалены в процессе подготовки набора данных для обучения (Рисунок 9).

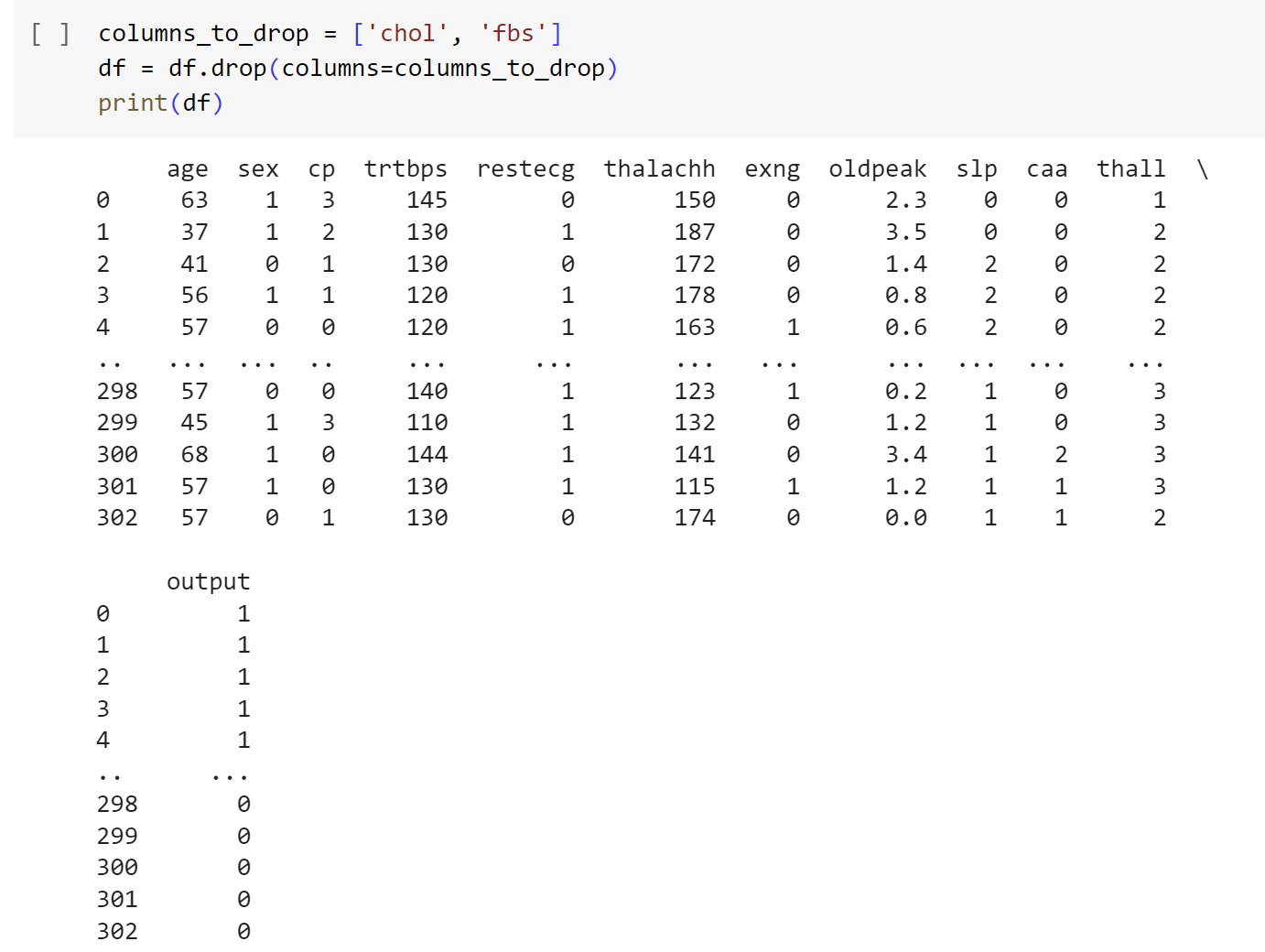


Рисунок 9 - Удаление столбцов chol и fbs

# Логистическая регрессия

При реализации метода логистической регрессии были использованы библиотеки sklearn и statmodels. Для построения, обучения и оценки модели были загружены необходимые функции из библиотеки sklearn (Рисунок 10).

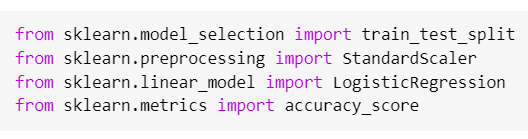


Рисунок 10 – Загрузка функций из библиотеки sklearn

Выполнили разделение предобработанных данных на признаки и целевую переменную, а также на обучающую и тестовую выборки. В нашем случае 20% данных будет отложено для тестирования, а 80% будут использованы для обучения модели.

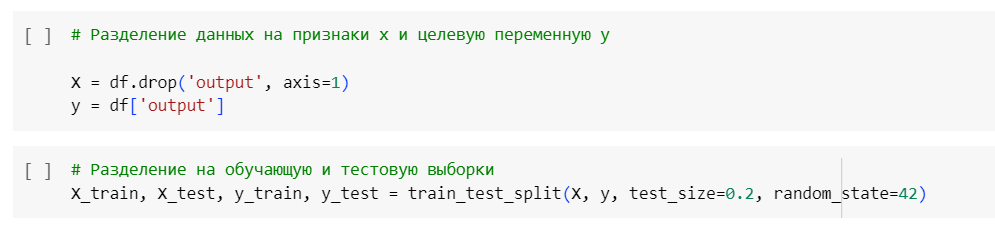


Рисунок 11 – Разделение данных на признаки и целевую переменную

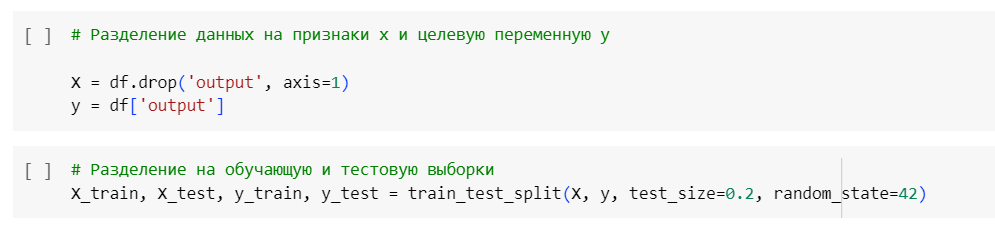


Рисунок 12 – Разделение на обучающую и тестовую выборки

Для предварительного анализа модели загрузим необходимые пакеты из библиотеки statsmodels. Далее построим модель и проведем диагностику.

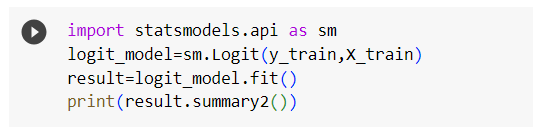


Рисунок 13 – Обучение модели и получение сводки

С помощью функции summary получили подробное описание данных и модели, а также коэффициенты модели и z-статистику.

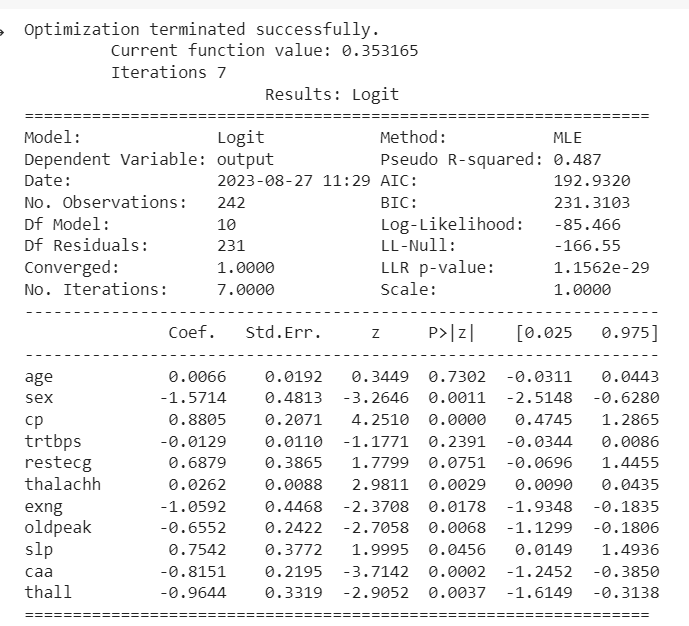


Рисунок 14 – Обучение модели и получение сводки

При анализе полученных данных было принято решение постепенно исключать переменные, p-значения которых больше 0,05, так как они являются незначимыми.

После исключения признаков age, trtbps, restecgможно заметитьуменьшение критериев качества AIC и BIC и увеличение абсолютного значения критерия Log-likelihood, что свидетельствует об улучшении модели (Рисунок 15).

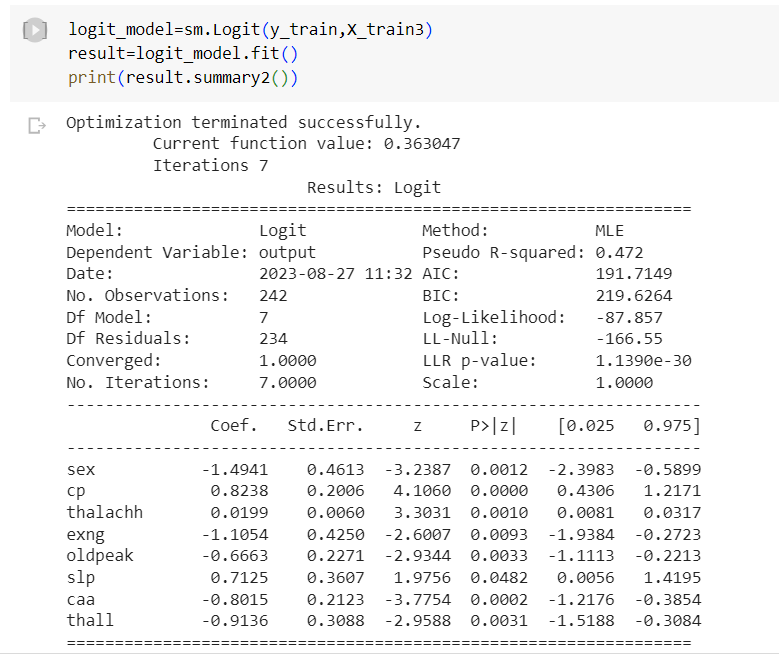


Рисунок 15 – Модель после удаления признаков

Выполним стандартизацию признаков и приступим к созданию и обучению модели логистической регрессии.

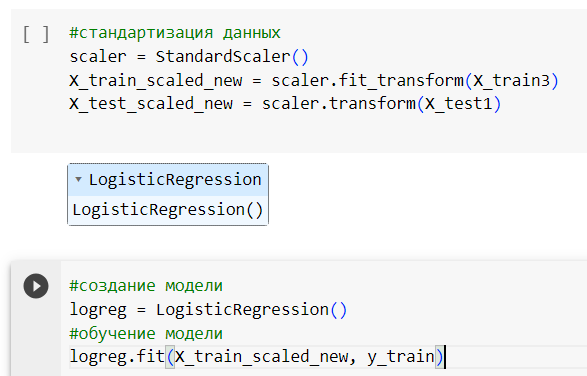


Рисунок 16 – Стандартизация признаков

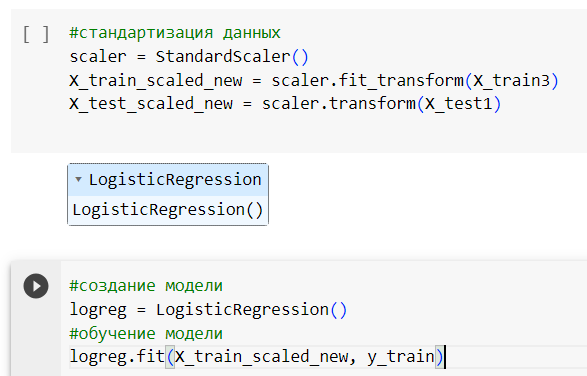


Рисунок 17 – Создание и обучение модели

Далее выполнили прогноз на тестовом наборе и провели расчёт точности модели.

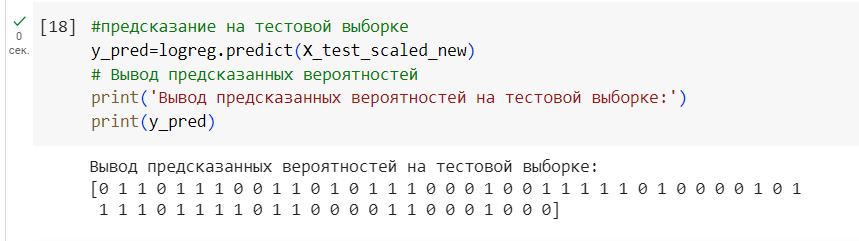


Рисунок 18 – Предсказание на тестовой выборке

Точность модели составила 0,869, что является хорошим показателем.

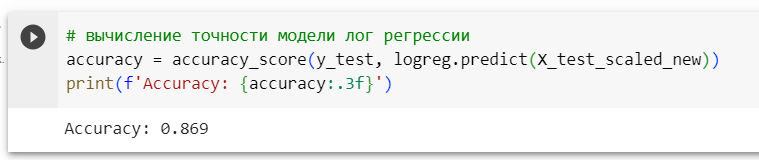


Рисунок 19 – Оценка точности модели

При анализе построенной матрицы путаницы выяснили, что из 61 спрогнозированного значения 53 прогноза оказались верными, а 8 неверными.

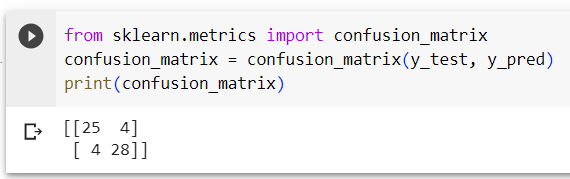


Рисунок 20 – Матрица путаницы

С помощью функции classification\_report получили показатели для оценки качества модели:

**1. Точность**: процент правильных положительных прогнозов по отношению к общему количеству положительных прогнозов.

**2. Полнота**: процент правильных положительных прогнозов по отношению к общему количеству фактических положительных результатов.

**3. Оценка F1** : средневзвешенное гармоническое значение точности и полноты. Чем ближе к 1, тем лучше модель.

В нашем случае показатели близки к единице, что свидетельствует о приемлемом качестве модели.

Так же стоит отметить, что такие показатели, как **чувствительность** в нашем случае составляет 0,86, а **специфичность** 0,88. Как известно, модель с высокой специфичностью чаще дает истинный результат при наличии отрицательного исхода (обнаруживает отрицательные примеры).

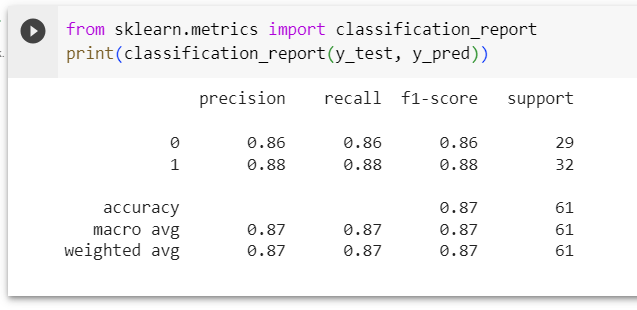


Рисунок 21 – Показатели качества

Для представления результатов бинарной классификации выполним построение ROC-кривой.

ROC-кривая показывает зависимость количества верно классифицированных положительных примеров от количества неверно классифицированных отрицательных примеров.

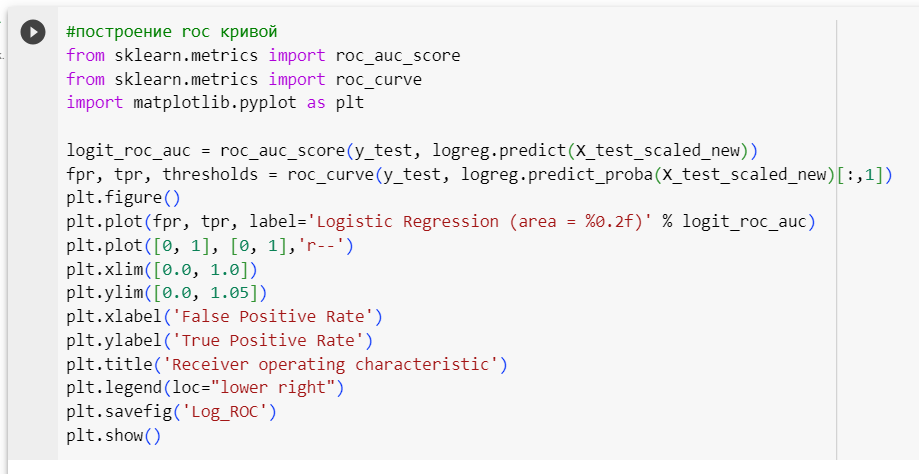


Рисунок 22 – Построение ROC-кривой с помощью функции roc\_curve

При визуальном анализе определено, что построенная ROC-кривая значительно охватывает левый верхний угол (Рисунок 23), что свидетельствует о большой предсказательной способности модели.

Значение площади под кривой – **AUC** равно 0,87, что говорит об очень хорошем качестве модели.

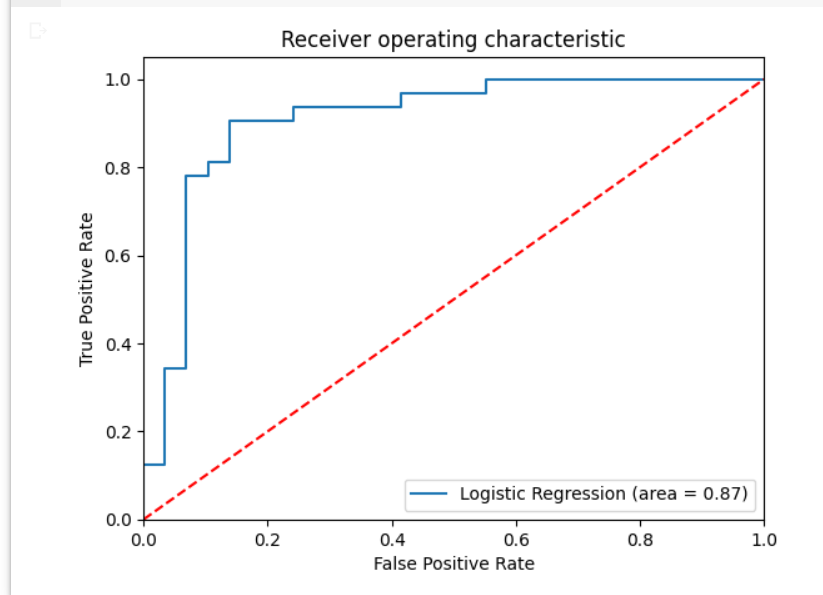


Рисунок 23 –ROC-кривая метода логистической регрессии

Также выведем коэффициенты регрессионного уравнения модели и значение R-квадрата.

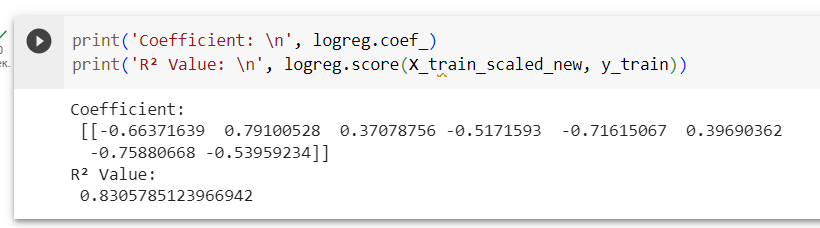


Рисунок 24 –Коэффициенты регрессионного уравнения и значение R2

# Метод k-ближайших соседей

Для реализации метода k-ближайших соседей была задействована библиотека sklearn для построения, обучения и оценки моделей машинного обучения, а также для предварительной обработки данных и оценки их производительности. Установлены модули для разделения, стандартизации набора данных, для реализации алгоритма k-NN и оценки точности модели (Рисунок 25).

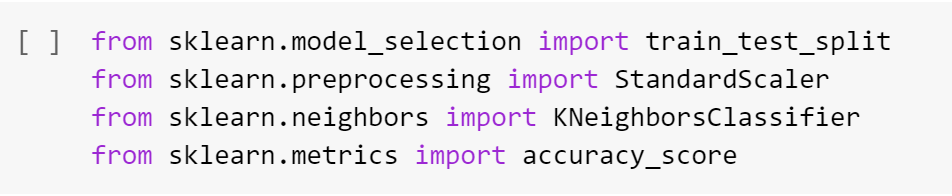


Рисунок 25 - Установка библиотек для метода k-NN

На рисунке 24 набор данных после предобработки разделен на признаки и целевую переменную, а также на обучающую и тестовую выборки. В нашем случае 20% данных будет отложено для тестирования, а 80% будут использованы для обучения модели. После стандартизации признаков реализован метод k-ближайших соседей.

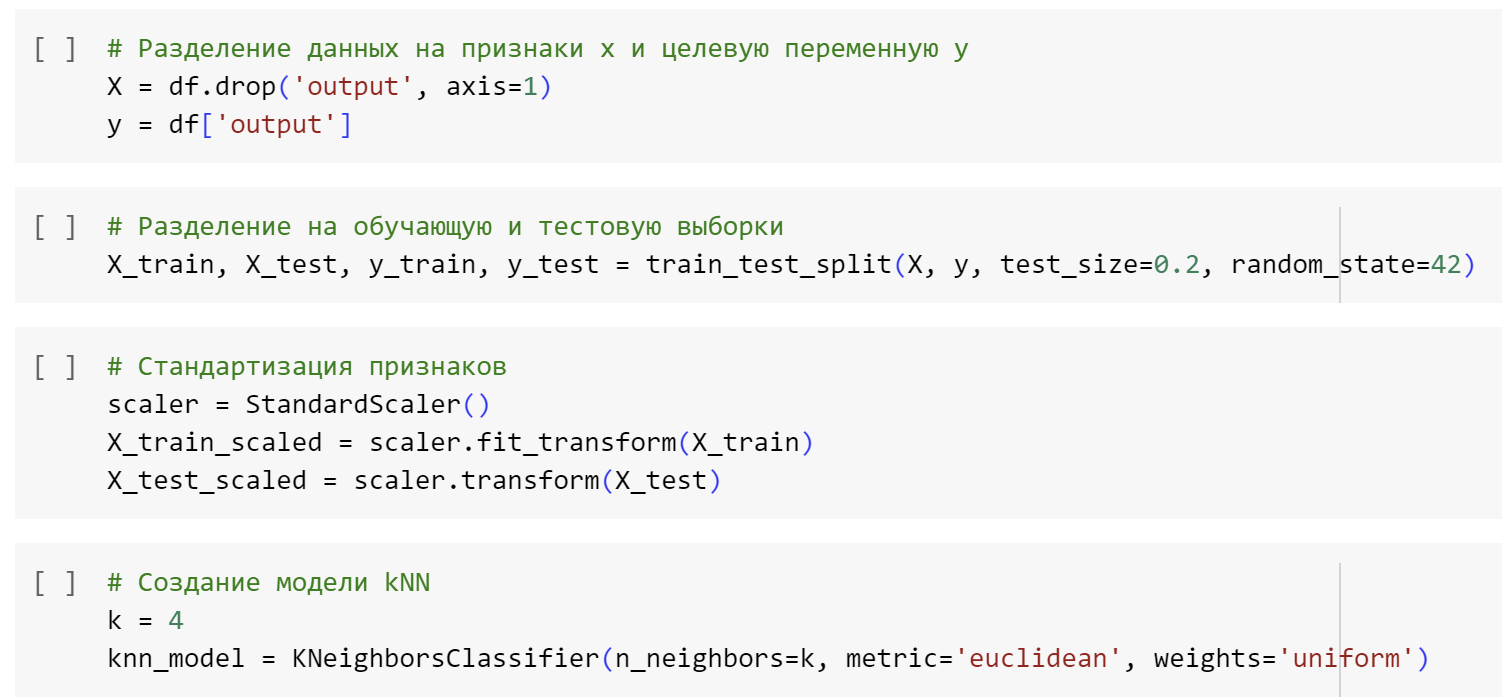


Рисунок 26 - Разбиение выборки для обучения модели

После обучения модели рассмотрено предсказание вероятности возникновения сердечного приступа на тестовой выборке (Рисунок 27).

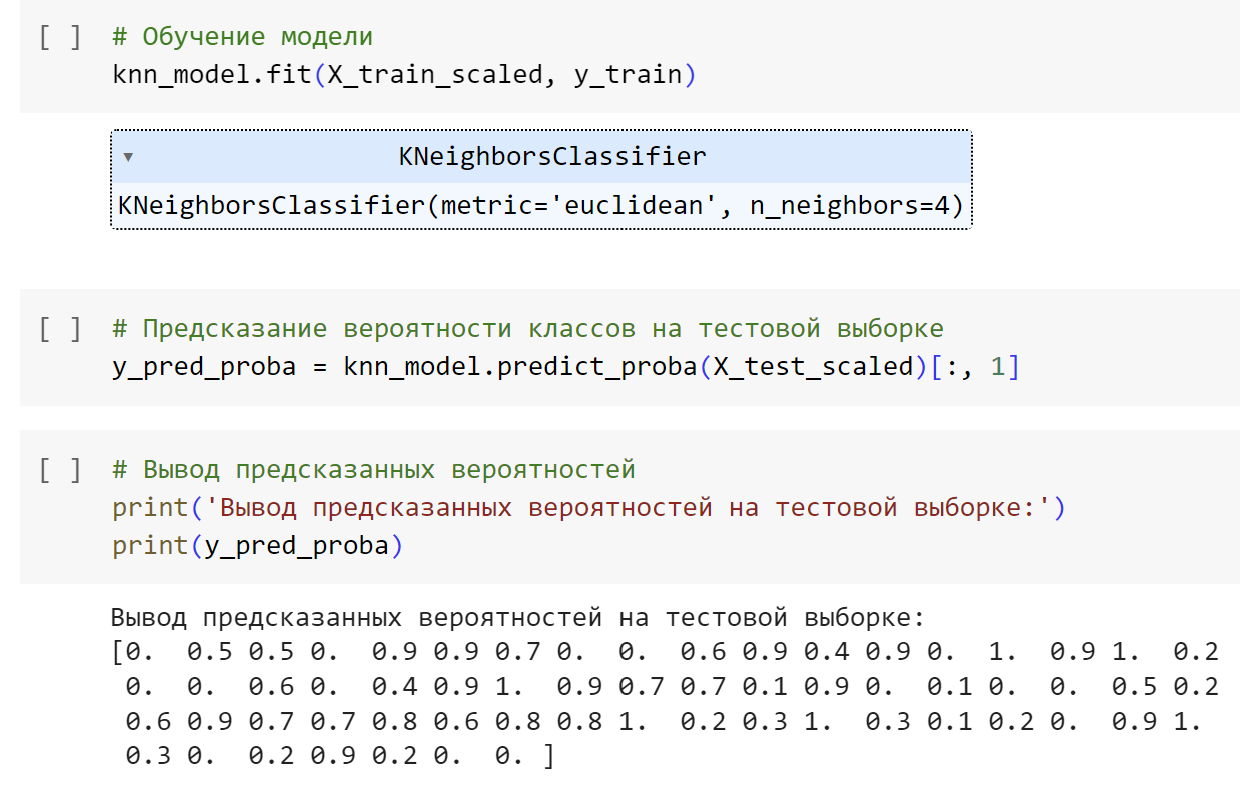


Рисунок 27 - Обучение модели k-NN

Были использованы кросс-валидация и различные значения числа соседей для оценки производительности модели на разных наборах данных, чтобы определить оптимальное количество соседей для нашей модели. Выбор параметра cv (количество фолдов) может варьироваться в зависимости от задачи и объема данных, и должен быть таким, чтобы оценить производительность модели надежно, но не сделать кросс-валидацию слишком медленной или затратной по времени. Это компромисс между точностью оценки и временем выполнения. Для нашего набора данных мы начали с cv=5 до cv=10, что дало разные результаты точности модели (Рисунок 28-Рисунок 31).

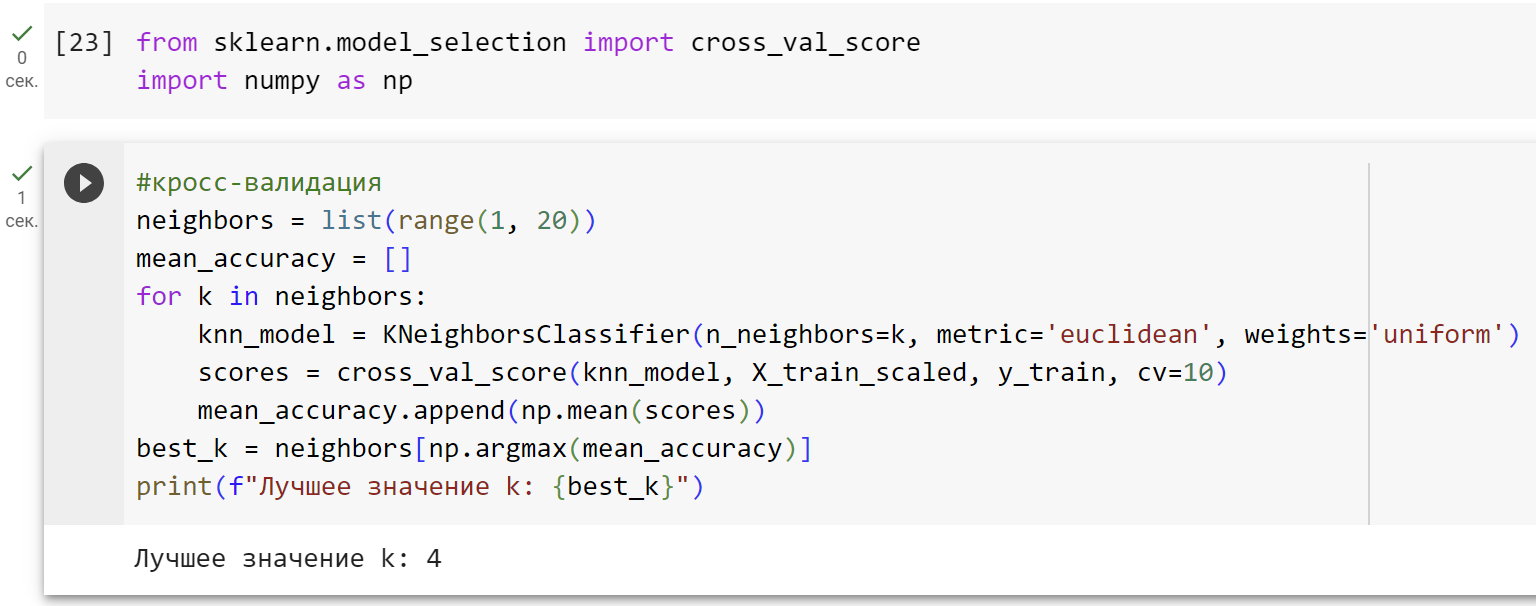


Рисунок 28 - Результат кросс-валидации при количестве фолдов, равным 10



Рисунок 29 - Точность модели при k=4

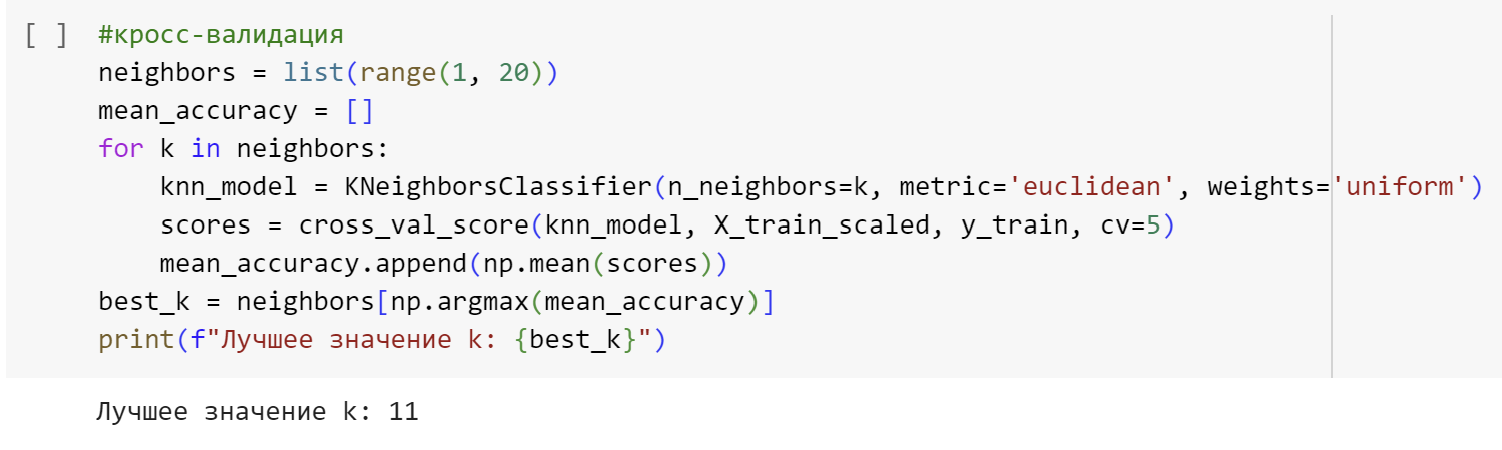


Рисунок 30 - Результат кросс-валидации при количестве фолдов, равным 5

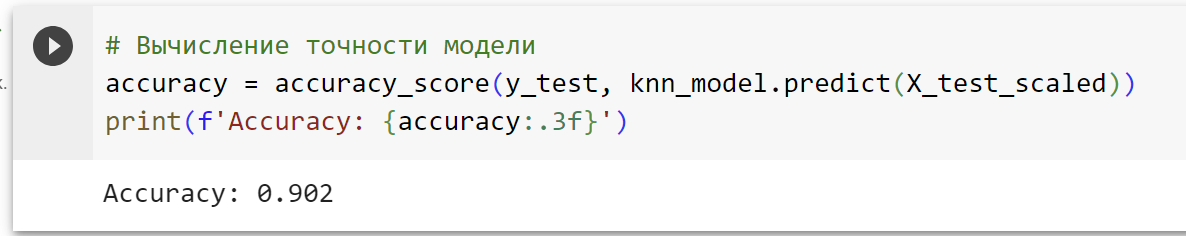


Рисунок 31 - Точность модели при k=11

Меньшее количество соседей в методе k-NN (меньше 5) делает модель более чувствительной к мелким нюансам в данных и адаптированной к сложным структурам данных, однако это может сделать ее более подверженной шуму в данных и более склонной к переобучению, особенно если данных мало.

Большое количество соседей (более 20) делает модель менее склонной к переобучению и более устойчивой к выбросам в данных, но также менее чувствительной к структуре данных и менее способной к выявлению важных закономерностей.

При задаче максимизации точности модели k=11 с точностью 0.902 предпочтительнее k=4 с точностью 0.803.

Для малых наборов данных также лучше использовать большее значение k, чтобы избежать переобучения, а для больших наборов данных можно использовать меньшее значение, чтобы учесть более мелкие детали данных. Таким образом, использование алгоритма k-NN при k=11 является оптимальным выбором.

В ходе анализа построенной матрицы путаницы выявили, что из 61 спрогнозированного значения 55 прогноза оказались верными, а 6 неверными (Рисунок 32).

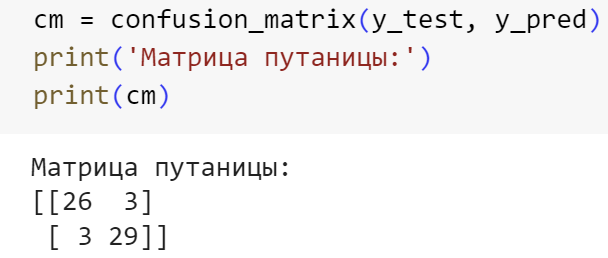


Рисунок 32 – Матрица путаницы для метода k-ближайших соседей

С помощью функции classification\_report получили показатели для оценки качества модели. В нашем случае показатели стремятся к единице, что свидетельствует о приемлемом качестве модели.

Так же стоит отметить, что такие показатели, как **чувствительность** в нашем случае составляет 0.9, а **специфичность** 0.91 (Рисунок 33). Как известно, модель с высокой специфичностью чаще дает истинный результат при наличии отрицательного исхода (обнаруживает отрицательные примеры).

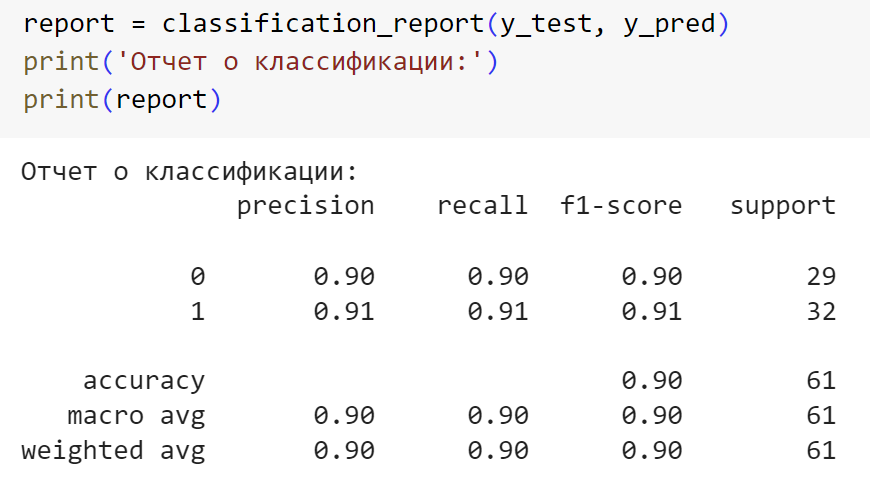


Рисунок 33 – Показатели качества модели k-NN

Для представления результатов бинарной классификации выполним построение ROC-кривой. При визуальном анализе определено, что построенная ROC-кривая значительно охватывает левый верхний угол, что свидетельствует о большой предсказательной способности модели.

Значение площади под кривой – **AUC** равно 0.91, что говорит об отличном качестве модели (Рисунок 34).

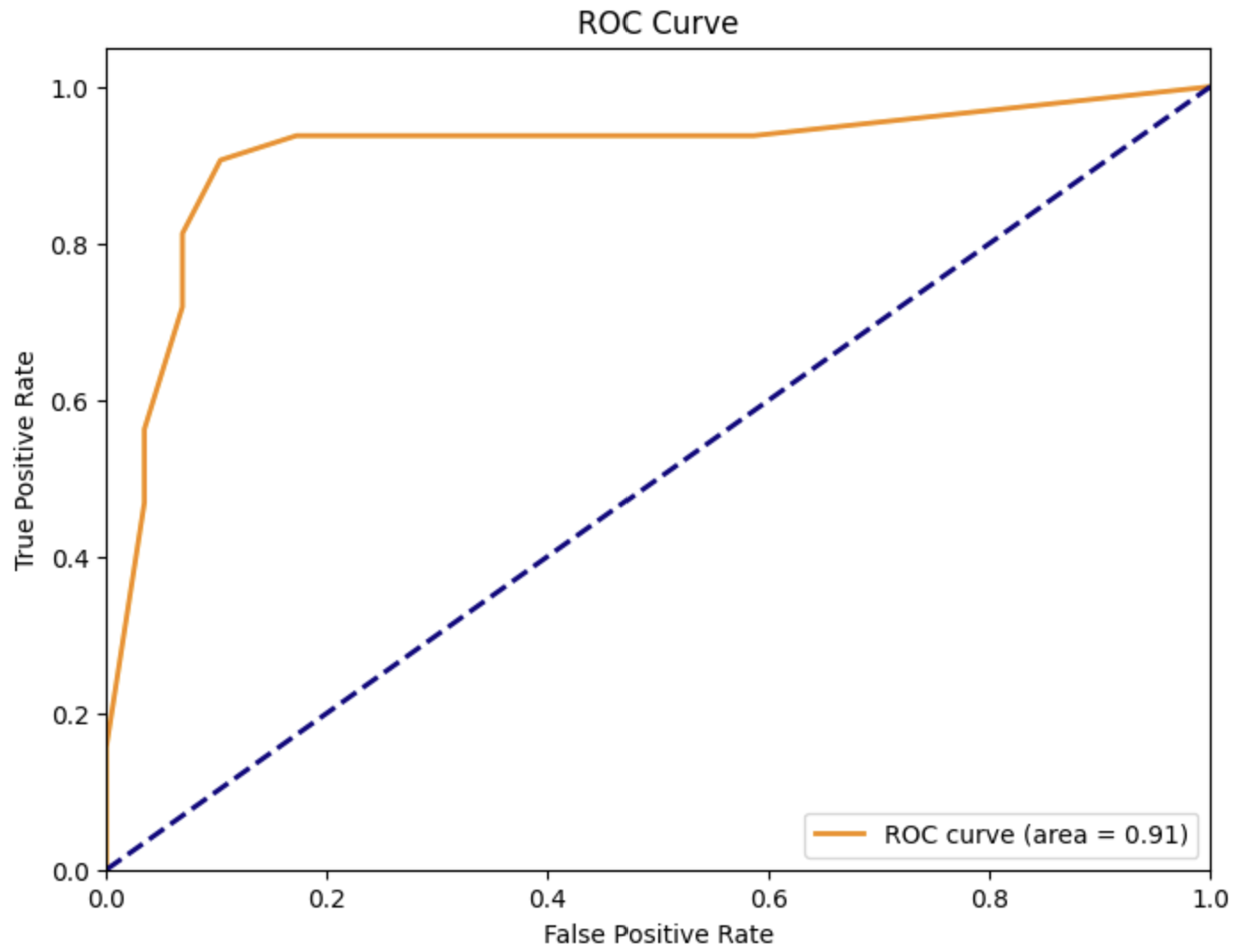


Рисунок 34 –ROC-кривая модели k-NN

# Дерево решений

При реализации метода дерева решений для построения, обучения и оценки модели были загружены необходимые функции из библиотеки sklearn (Рисунок 35).

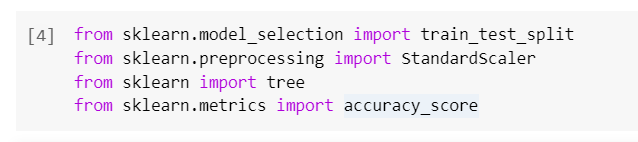


Рисунок 35 – Загрузка функций из библиотеки sklearn

Выполнили разделение предобработанных данных на признаки и целевую переменную, а также на обучающую и тестовую выборки. В нашем случае 20% данных будет отложено для тестирования, а 80% будут использованы для обучения модели.

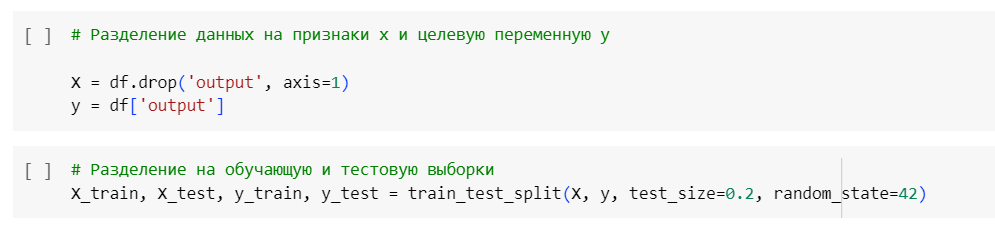


Рисунок 36 – Разделение данных на признаки и целевую переменную

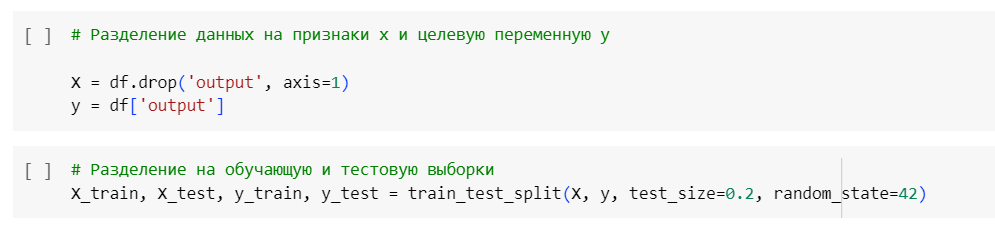


Рисунок 37 – Разделение на обучающую и тестовую выборки

Выполним стандартизацию признаков и приступим к созданию и обучению модели decision tree.

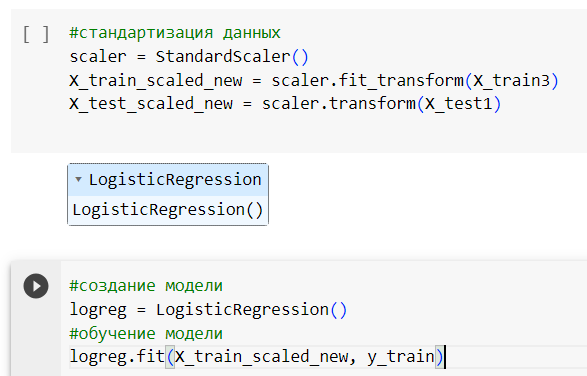


Рисунок 38 – Стандартизация признаков

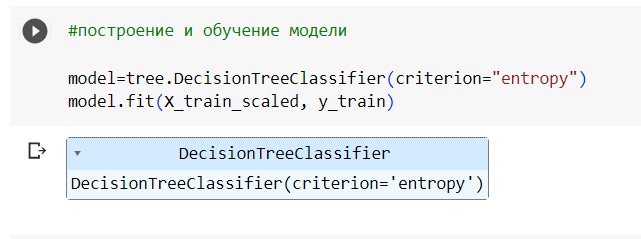


Рисунок 39 – Создание и обучение модели

Далее выполнили прогноз на тестовом наборе и провели расчёт точности модели.

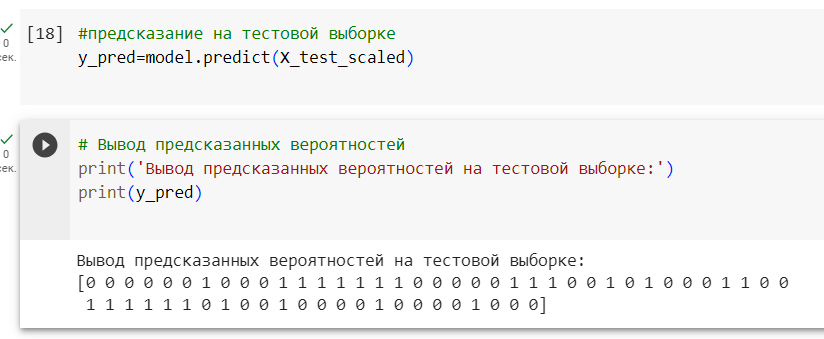


Рисунок 40 – Предсказание на тестовой выборке

Точность модели составила 0,82 (Рисунок 41), что значительно ниже по сравнению с моделью логистической регрессии и моделью k-NN.

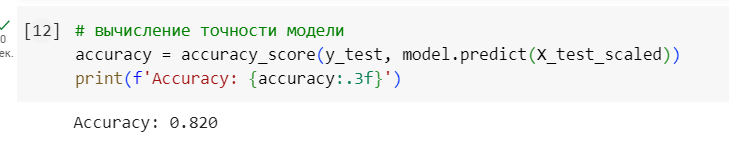


Рисунок 41 – Оценка точности модели

При анализе построенной матрицы путаницы выяснили, что из 61 спрогнозированного значения 50 прогнозов оказались верными, а 11 неверными.

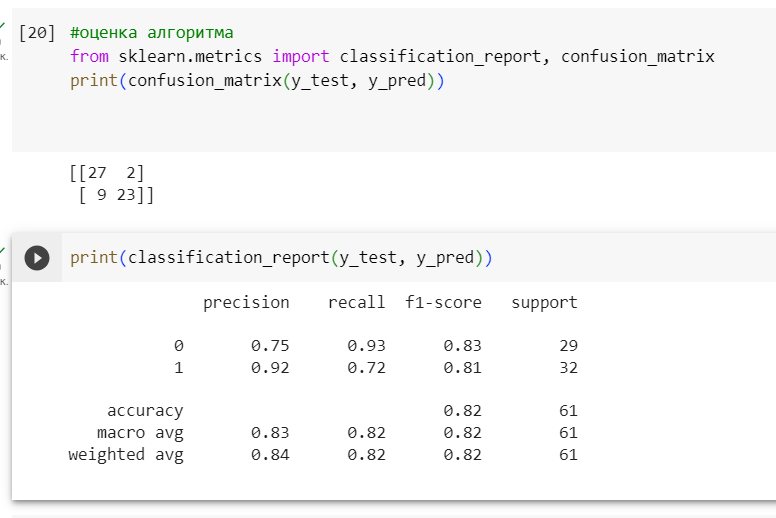


Рисунок 42 – Матрица путаницы

С помощью функции classification\_report получили показатели для оценки качества модели. **Чувствительность** в нашем случае составляет 0,75, а **специфичность** 0,92. Следовательно, модель дерева решений чаще дает истинный результат при наличии отрицательного исхода (обнаруживает отрицательные примеры).

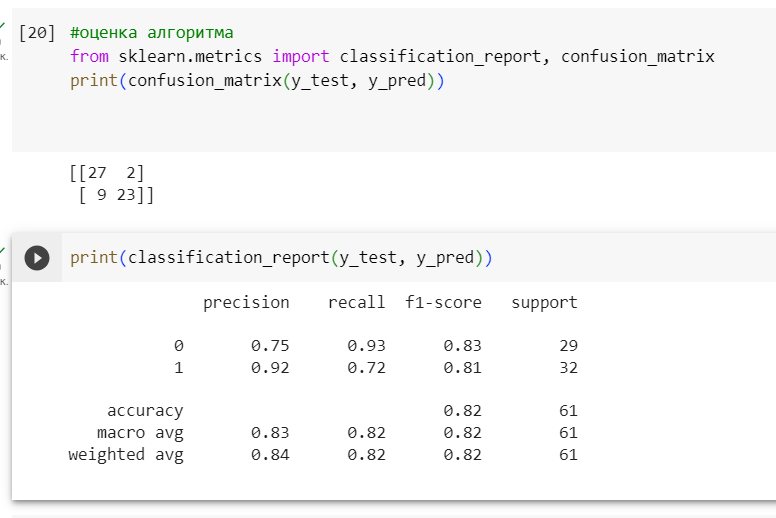


Рисунок 43 – Показатели качества

Для представления результатов бинарной классификации выполним построение ROC-кривой.

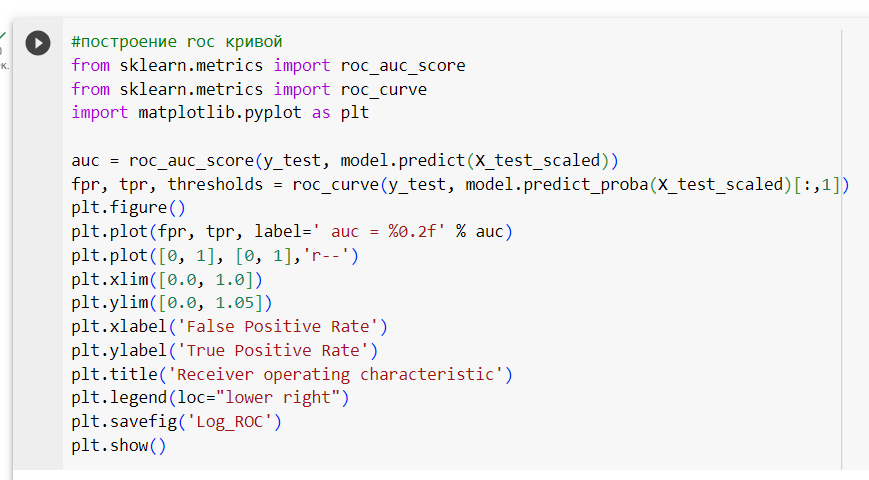


Рисунок 44 – Построение ROC-кривой с помощью функции roc\_curve

Значение площади под кривой – **AUC** равно 0,88, что говорит о хорошем качестве модели (Рисунок 45).

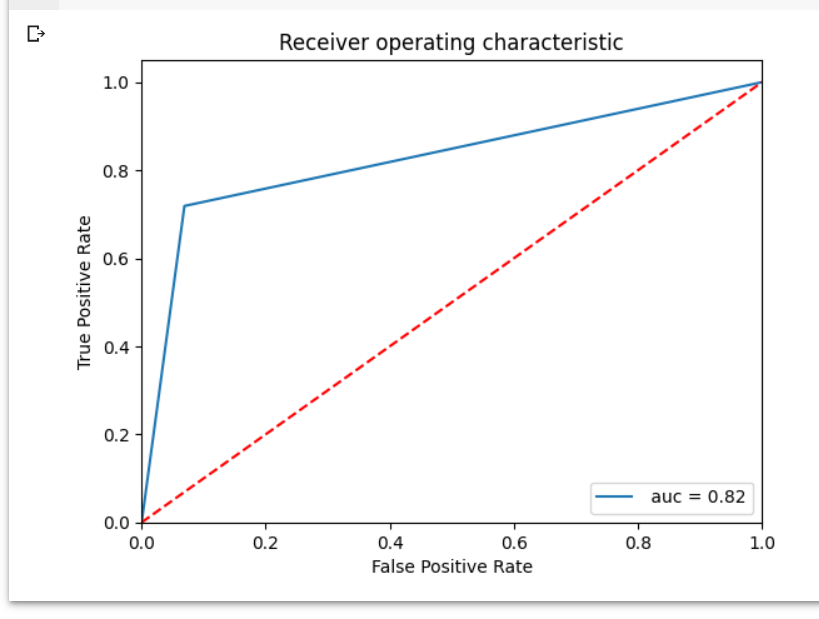


Рисунок 45 –ROC-кривая модели дерева решений

# Анализ методов и алгоритмов обучения

Для классификационных моделей, как и для моделей регрессии, актуальна задача оценки их качества для определения работоспособности моделей и их сравнения. Однако решение этой задачи для моделей классификации вообще, и бинарной классификации в частности, сложнее, чем для регрессии. Связано это с тем, что целевая переменная является категориальным (дискретным) значением, и, следовательно, ошибка классификации не может быть выражена числовым значением.

Поэтому в основе оценки качества классификационных моделей лежит статистика результатов классификации обучающих примеров. С ее помощью вычисляются метрики качества — показатели, которые зависят от результатов классификации и не зависят от внутреннего состояния модели.

В нашем случае для оценки качества рассматриваемых моделей (логистическая регрессия, k-ближайших соседей и дерево решений) будем использовать следующие метрики:

1. Матрица ошибок (Сonfusion matrix).
2. Меткость (Accuracy).
3. Точность (Precision).
4. Чувствительность (Recall).
5. Специфичность (Specificity).
6. F1-мера (F1-score).
7. Площадь под ROC-кривой (Area under ROC-curve, AUC-ROC).

При анализе матриц ошибок выявлено, что у модели k-ближайших соседей оказалось наибольшее количество верно классифицированных значений (из 61 значения тестовой выборки верно классифицировала 55 значений и 6 неверно). Соответственно, доля правильно классифицированных значений Accuracy у модели k-ближайших соседей наибольшая и составляет 0,9. Но стоит отметить, что данная метрика бесполезна в задачах с неравными классами, поэтому необходимо проанализировать и другие.

Точность равна доле истинноположительных классификаций к общему числу положительных классификаций.

Наибольшее значение Precision, равное 0,93, показала модель дерева решений.

Полнота, известная еще как чувствительность, определяется как число истинноположительных классификаций относительно общего числа положительных наблюдений.

Наибольшее значение Recall оказалось у модели k-ближайших соседей и составило 0,9. Следовательно, данная модель часто дает истинный результат при наличии положительного исхода (обнаруживает положительные примеры).

Специфичность классификатора — это доля истинноотрицательных классификаций в общем числе отрицательных классификаций:

Наибольшее значение Specificity оказалось у модели дерева решений и составило 0,92. Следовательно, данная модель чаще дает истинный результат при наличии отрицательного исхода (обнаруживает отрицательные примеры).

Чем выше точность и полнота, тем лучше модель. Но на практике их максимальные значения одновременно недостижимы, поэтому приходится искать баланс между ними. Для этого используется метрика, объединяющая в себе информацию о точности и полноте - F1-мера. Наибольшее значение F1-меры показала модель k-ближайших соседей (0,9), что указывает на её высокую эффективность.

При визуальной оценке ROC-кривых трех методов определено, что кривая модели k-NN, расположена выше и левее по сравнению с остальными, что свидетельствует о большей предсказательной способности модели. Однако, визуальное сравнение кривых ROC не всегда позволяет выявить наиболее эффективную модель, поэтому необходимо еще оценить и площадь под кривой ROC.

Численный показатель площади под кривой называется AUC и часто используется для сравнительного анализа нескольких моделей. У модели k-ближайших соседей показатель AUC является наибольшим (0,91), что говорит о прогностической силе модели.

В таблицу 1 занесли полученные значения метрик качества для трёх используемых методов.

Таблица 1 – Сравнение моделей

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Accuracy | Precision | Recall | Specificity | F1-score | AUC |
| Логистическая регрессия | 0,87 | 0,86 | 0,86 | 0,88 | 0,86 | 0,87 |
| Метод k-ближайших соседей | 0,9 | 0,9 | 0,9 | 0,91 | 0,9 | 0,91 |
| Дерево решений | 0,82 | 0,93 | 0,75 | 0,92 | 0,83 | 0,82 |

Таким образом, при общем анализе метрик качества трёх рассматриваемых моделей можно сделать вывод, что наилучшие показатели наблюдаются у модели k-ближайших соседей. Следовательно, данная модель является наиболее эффективной для решения поставленной задачи – прогнозирования сердечного приступа.

# Заключение

В процессе выполнения итогового проекта по дисциплине «Технологии обработки количественных данных» было разработано программное решение по прогнозированию возможного наступления сердечного приступа у пациента.

В рамках проделанной работы был найден датасет содержащий необходимые для заявленной цели данные, проведенный интеллектуальный анализ которого позволил выявить ряд закономерностей, которые выделяли потенциально наиболее результативные методы решения задачи. После проведён сравнительный анализ некоторых алгоритмов и методов машинного обучения, из числа которых были отобраны три наиболее подходящих для имеющихся данных подхода, с точки зрения их объема, специфики и количества.

Каждый из трех методов был реализован в виде программы, для которых, помимо прочего, был собран одинаковый набор метрик для проведения сравнительной оценки качества разработанных решений, а именно: Матрица ошибок (Сonfusion matrix), Меткость (Accuracy), Точность (Precision), Чувствительность (Recall), Специфичность (Specificity), F1-мера (F1-score), Площадь под ROC-кривой (Area under ROC-curve, AUC-ROC).

Таким образом, посредством сравнения характеристик реализованных моделей, в рамках проведенного исследования, для данного датасета можно заявлять о более высокой прогностической силе у модели k-ближайших соседей. Соответственно, данная модель рекомендуется для использования в качестве системы поддержки принятия решения для специалиста при первичной оценке состояния пациента, направленного на кардиологическое исследование.

В качестве направления для дальнейшего развития исследования может быть заявлено расширение имеющегося набора данных и выявление дополнительных факторов влияния на исследуемое состояние. Работа в данных направлениях может осуществляться с целью перехода на менее склонные к переобучению методы машинного обучения.