МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего

образования

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Отчет по заданию №1

по дисциплине

«ГЛУБИННОЕ ОБУЧЕНИЕ»

**«Задание №1 – Полносвязные сети прямого распространения»**

Выполнил

Студент гр. 932226

\_\_\_\_\_\_\_\_\_\_\_Савенкова М. М. «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г

Проверил

\_\_\_\_\_\_\_\_ к.т.н. Аксёнов С.В.

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2023 г.

Томск – 2023

Оглавление

[1 Индивидуальное задание 3](#_Toc151043208)

[2 Цель работы 4](#_Toc151043209)

[3 Листинг программ 4](#_Toc151043210)

[4 Построение моделей 5](#_Toc151043211)

[4.1 Бинарный классификатор 5](#_Toc151043212)

[4.1.1 Входные данные 5](#_Toc151043213)

[4.1.2 Предобработка и разведывание данных 8](#_Toc151043214)

[4.1.3 Построенные нейросетевые модели 18](#_Toc151043215)

[4.1.4 Выводы по задаче бинарной классификации 30](#_Toc151043216)

[4.2 Многоклассовый классификатор 30](#_Toc151043217)

[4.2.1 Входные данные 30](#_Toc151043218)

[4.2.2 Предобработка и разведывание данных 32](#_Toc151043219)

[4.2.3 Построенные нейросетевые модели 43](#_Toc151043220)

[4.2.4 Выводы по задаче многоклассовой классификации 57](#_Toc151043221)

[4.3 Нейросетевой регрессор 57](#_Toc151043222)

[4.3.1 Входные данные 57](#_Toc151043223)

[4.3.2 Предобработка и разведывание данных 57](#_Toc151043224)

[4.3.3 Построенные нейросетевые модели 64](#_Toc151043225)

[4.3.4 Выводы по задаче регрессии 72](#_Toc151043226)

[5 Выводы 73](#_Toc151043227)

# 1 Индивидуальное задание

Используя язык программирования Python, построить три нейросетевые модели:

1. Бинарный Классификатор

Оценка вероятности, того, что клиент откроет банковский депозит в результате маркетинговой акции: <https://archive.ics.uci.edu/ml/datasets/Bank+Marketing>

Класс: атрибут 21 - y - has the client subscribed a term deposit? (binary: 'yes', 'no').

Вычислить следующие метрики работы: Recall, Precision, Weighted Accuracy, AUC для всех исследованных моделей.

1. Многоклассовый классификатор

Оценка здоровья внутриутробного развития плода: <https://www.kaggle.com/datasets/andrewmvd/fetal-health-classification>

Метка класса: fetal\_health.

Вычислить следующие метрики работы: Recall, Precision, Weighted Accuracy, AUC для всех классов всех исследованных моделей. Вывести ROC-кривые для каждого класса в лучшем классификаторе.

1. Регрессор

Оценка уровня потребления ресурсов домохозяйством. Набор данных DS\_2019\_public.csv. Описание атрибутов – файл recs2009\_public\_codebook.xlsx.

Предсказываемый признак – один из атрибутов (использовать только один из): TOTALBTU, TOTALBTUSPH, TOTALBTUCOL, TOTALBTUWTH, TOTALBTURFG, TOTALBTUOTH (Примечание BTU – Британская тепловая единица BTU ≈ 252 калории).

Вычислить следующие метрики работы: MSE, MAE, R2 для всех полученных моделей.

Для каждого задания необходимо выполнить загрузку и предварительную обработку данных из наборов, разделить каждую выборку на обучающую, тестовую и валидационную и произвести обучение набора нейросетевых архитектур, отличающихся разным набором параметров: число слоёв, количество нейронов в слоях, функции активации в слоях, процедур оптимизации: то есть подобрать архитектуры нейронных сетей, которые с одной стороны позволяют получить модели с лучшими метриками качества работы, с другой стороны не являются избыточными и не переобученными. Сделать выводы по результатам построения моделей.

# 2 Цель работы

Получить навыки создания полносвязного бинарного и многоклассового классификаторов, а также нейросетевого регрессора с помощью библиотеки построения нейронных сетей.

# 3 Листинг программ

Листинги программ представлены в отдельных файлах:

1. Бинарный классификатор ‒ Бинарный\_классификатор.ipynb
2. Многоклассовый классификатор ‒ Многоклассовый\_классификатор.ipynb
3. Регрессор ‒ Регрессор.ipynb

# 4 Построение моделей

4.1 Бинарный классификатор

4.1.1 Входные данные

Данные связаны с кампаниями прямого маркетинга (телефонными звонками) португальского банковского учреждения. Цель классификации состоит в том, чтобы предсказать, подпишется ли клиент на срочный депозит (переменная y).

Маркетинговые кампании были основаны на телефонных звонках. Часто требовалось более одного контакта с одним и тем же клиентом, чтобы получить доступ к тому, будет ли продукт (банковский срочный депозит) подписан ("да") или нет ("нет").

Существует четыре набора данных:

1. bank-additional-full.csv со всеми примерами (41188) и 20 входными данными, упорядоченными по дате (с мая 2008 по ноябрь 2010), очень близко к данным, проанализированным в [Moro et al., 2014]
2. bank-additional.csv с 10% примеров (4119), выбранных случайным образом из 1), и 20 входными данными.
3. bank-full.csv со всеми примерами и 17 входными данными, упорядоченный по дате (более старая версия этого набора данных с меньшим количеством входных данных).
4. bank.csv с 10% примеров и 17 входными данными, случайным образом выбранными из 3 (более старая версия этого набора данных с меньшим количеством входных данных).

Цель классификации состоит в том, чтобы предсказать, подпишется ли клиент (да/нет) на срочный депозит (переменная y).

В данной работе был использован датасет bank-additional.csv. В наборе данных содержатся следующие признаки:

1. Данные клиента банка

* age (возраст)
* job (работа) – тип работы; категориальный;
* marital (семейное положение) – статус семейного положения; категориальный;
* education (образование) – тип образования; категориальный;
* default (дефолт) – есть ли кредит в состоянии дефолта, то есть кредит, по которому заемщик не исполняет или не в состоянии исполнить свои кредитные обязательства; категориальный;
* housing (жильё) – есть ли у клиента жилищный кредит; категориальный
* loan (кредит) – есть ли у клиента личный заём; категориальный.

1. Данные, связанные с последним контактом текущей кампании

* contact (контакт) – тип контактного общения; категориальный;
* month (месяц) – месяц последнего контакта в году; категориальный;
* day\_of\_week (день недели) – последний контактный день недели; категориальный;
* duration – продолжительность последнего контакта в секундах; числовой.

1. Другие признаки

* campaign (кампания) – количество контактов, выполненных в ходе этой кампании и для данного клиента; числовое значение, включает последний контакт;
* pdays – количество дней, прошедших после того, как с клиентом в последний раз связывались в рамках предыдущей кампании; числовое значение, 999 означает, что ранее с клиентом не связывались;
* previous – количество контактов, выполненных до этой кампании и для данного клиента; числовое значение;
* poutcome: результат предыдущей маркетинговой кампании; категориальный;

1. Атрибуты социального и экономического контекста

* emp.var.rate – коэффициент вариации занятости – квартальный показатель; числовой;
* cons.price.idx – индекс потребительских цен – месячный показатель; числовой;
* cons.conf.idx – индекс потребительского доверия – ежемесячный показатель; числовой;
* euribor3m – ставка euribor за 3 месяца – ежедневный индикатор; числовой
* nr.employed – количество сотрудников – квартальный показатель; числовой.

1. Целевая переменная

* y – оформил ли клиент срочный депозит; двоичный код: "да", "нет".

Данные представлены в виде таблицы. По строкам расположены

объекты – базовая единица данных; то, для чего необходимо выполнить

прогноз. Столбцы таблицы задают признаки объектов – характеристики,

позволяющие задавать особенности каждого конкретного объекта и отличать

объекты друг от друга.

* + 1. Предобработка и разведывание данных

**Проверка пропущенных значений**

Используя библиотеку missingno, убеждаемся, что в датасете нет пропусков (рисунок 1).

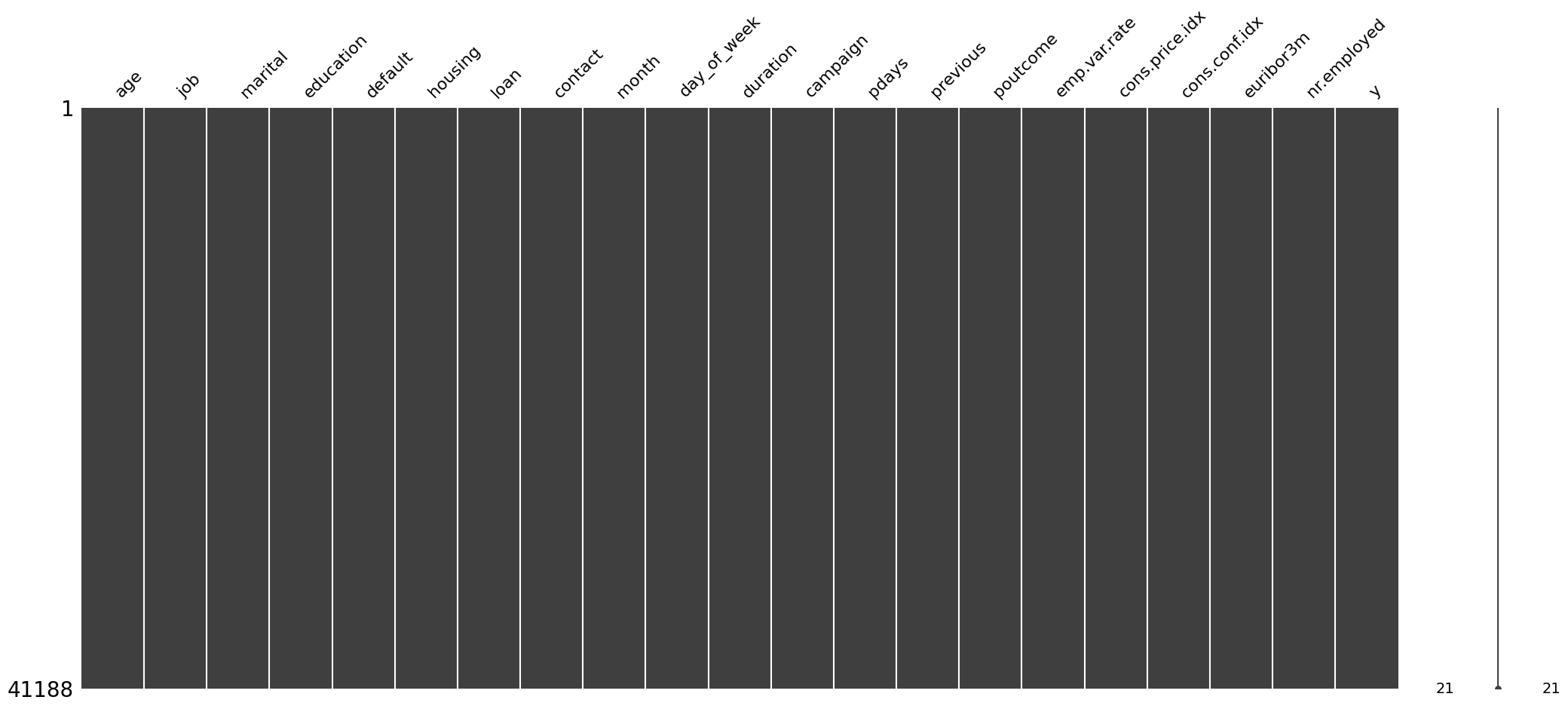


Рисунок 1 — Пропущенные значения

**Проверка на сбалансированность**

Распределение данных по классам целевой переменной (рисунок 2). По диаграмме можно судить о несбалансированности датасета.

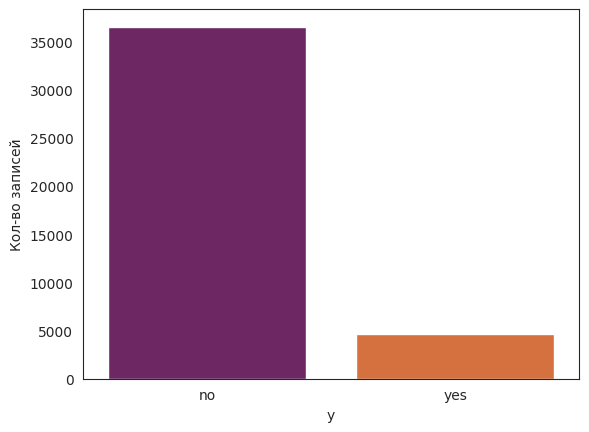


Рисунок 2 — Несбалансированность датасета

Несбалансированный датасет может привести к ряду проблем в задачах анализа данных и машинного обучения. Вот некоторые из потенциальных последствий несбалансированности датасета:

1. **Смещение модели:** Если классы в датасете представлены неравномерно, модель может быть смещена в сторону более представленного класса. В результате этого модель может показывать плохую производительность на менее представленных классах.
2. **Плохая обобщающая способность:** Модель, обученная на несбалансированных данных, может показывать плохую способность к обобщению на новых данных, особенно на данных из менее представленных классов.
3. **Плохая интерпретация результатов:** Несбалансированный датасет может привести к неправильной интерпретации результатов, особенно в задачах классификации. Метрики, такие как точность (accuracy), могут быть вводящими в заблуждение, так как модель может просто предсказывать доминирующий класс, не учась различать менее представленные классы.
4. **Неэффективное использование данных:** Модель может не эффективно использовать информацию из менее представленных классов, что может привести к потере ценных знаний и снижению производительности.
5. **Неадекватное обучение:** В случае крайней несбалансированности, модель может даже не смочь обучиться распознавать менее представленные классы из-за недостаточного числа примеров.

Для борьбы с несбалансированностью датасета существует несколько подходов, такие как увеличение (oversampling) менее представленных классов, уменьшение (undersampling) более представленных классов, использование весов классов в обучении модели и применение различных методов ансамблей.

При решении данной задачи для балансирования данных был использован метод undersampling.

**Матрица корреляции**

Матрица корреляции показывает степень линейной зависимости между парами признаков. Она измеряет, насколько близко связаны два признака друг с другом. Значения в матрице корреляции могут варьироваться от -1 до 1.

Когда значение корреляции близко к 1, это указывает на положительную линейную связь между двумя признаками. Это означает, что при увеличении одного признака значение второго признака также увеличивается.

Когда значение корреляции близко к -1, это указывает на отрицательную линейную связь между двумя признаками. Это означает, что при увеличении одного признака значение второго признака снижается.

Когда значение корреляции близко к 0, это указывает на отсутствие линейной связи между двумя признаками. Они независимы друг от друга.

На рисунке 3 представлена матрица корреляции между признаками данного набора данных. Как видим, в некоторых случаях корреляция между признаками достаточно высока.

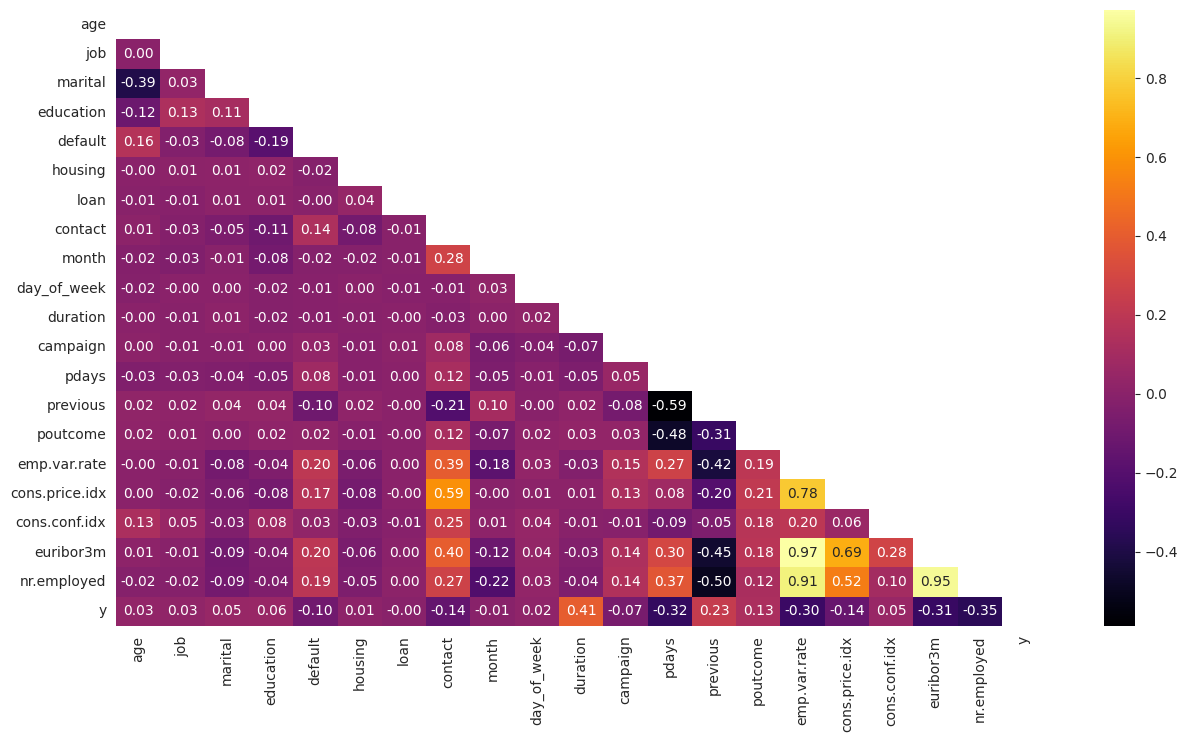


Рисунок 3 — Матрица корреляции

**Анализ зависимостей между признаками**

**Распределение возраста**

Построим возрастное распределение тех, кто оформил кредит, и тех, кто не оформил (рисунок 4).

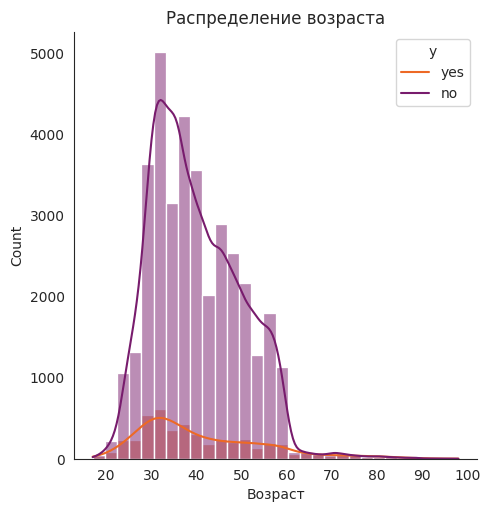


Рисунок 4 — Распределение возраста

Рассмотрим графики по отдельности (рисунок 5).

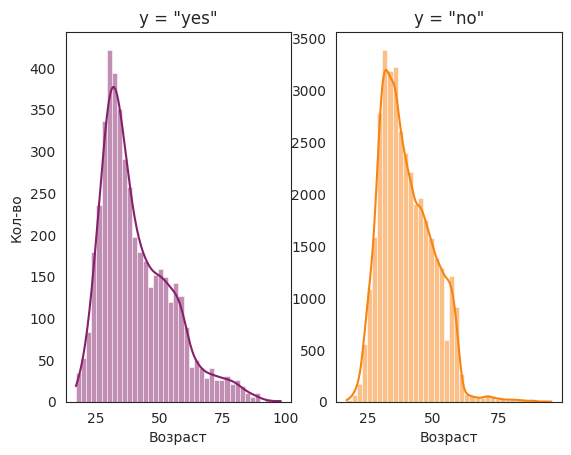


Рисунок 5 — Распределения возрастов на отдельных графиках

Как видим, графики очень похожи. Большинство клиентов, оформивших срочный депозит, находятся в возрасте от 30 до 50 лет, что также соответствует возрасту с наибольшим доходом.

**Зависимость целевой переменной от типа работы**

На рисунке 6 представлена зависимость целевой переменной от типа работы.



Рисунок 6 — Зависимость целевой переменной от типа работы

Пропорциональность согласившихся и отказавшихся примерно пропорциональна в каждом случае.

**Семейное положение**

На рисунке 7 представлена зависимость целевой переменной от типа семейного положения.

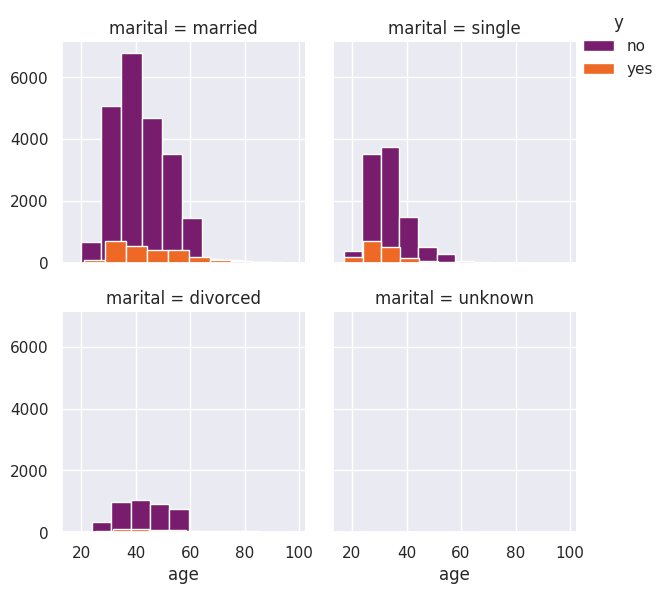


Рисунок 7 — Зависимость целевой переменной от типа семейного положения

Больше супружеских пар оформили срочный депозит.

**Образование**

Графики на рисунке 8 демонстрируют, что чем выше уровень образования человека, тем в более раннем возрасте он может получить кредит.

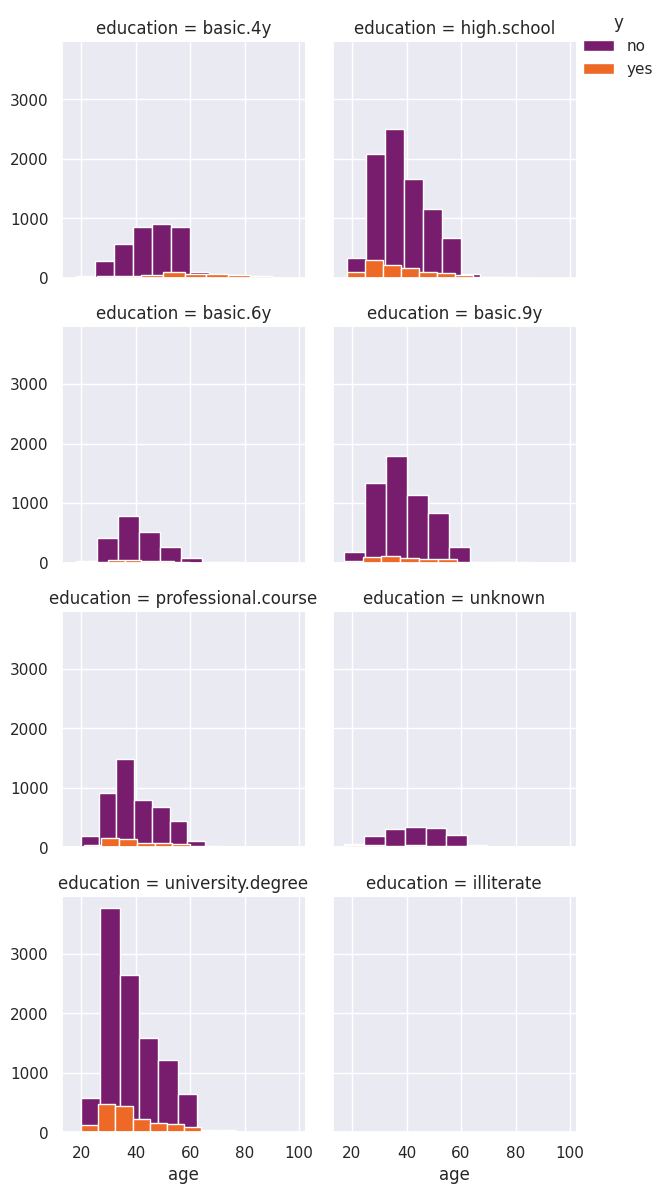
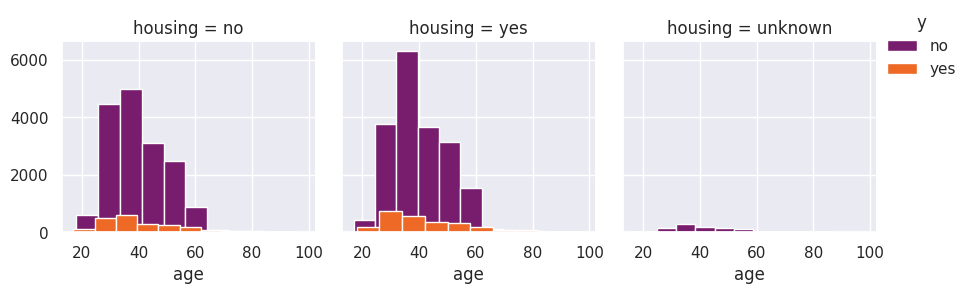


Рисунок 8 — Уровень образования

**Жилищный кредит**

Графики на рисунке 9 показывают, что наличие жилищного кредита слабо влияет на взятие кредита.

Рисунок 9 — Жилищный кредит

**Личный заём**

По графикам на рисунке 10 можно сделать заключение, что среди не имеющих личный заем вероятность взятия кредита больше.

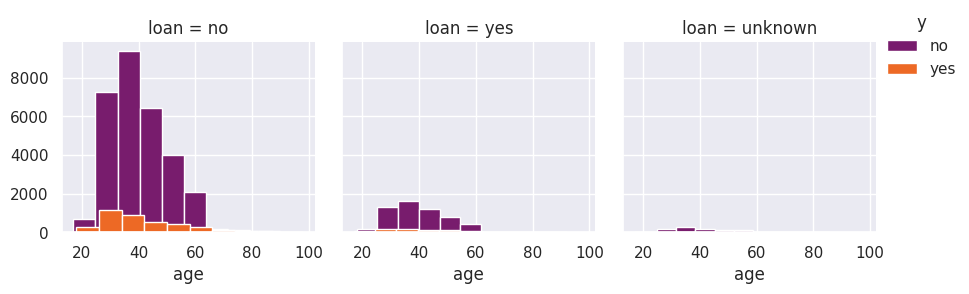


Рисунок 10 — Личный заем

**Использованные признаки**

Для анализа данных будем использовать все признаки из набора.

4.1.3 Построенные нейросетевые модели

4.1.3.1 Модель №1

**Параметры архитектуры**

Архитектура нейронной сети состоит из одного скрытого слоя с 32 нейронами и функцией активации ReLU, а также выходного слоя с одним нейроном и функцией активации sigmoid.

Функция активации sigmoid обычно используется в выходном слое нейросетей для задач бинарной классификации, где требуется предсказать один бинарный результат. Эта функция переводит входные значения в диапазон между 0 и 1, что удобно для интерпретации как вероятности принадлежности к классу. Если выходное значение близко к 1, то модель предсказывает, что объект принадлежит к данному классу, а если близко к 0, то объект не принадлежит.

Функция активации ReLU обычно используется во внутренних слоях нейронных сетей для задач классификации, регрессии и других задач машинного обучения. Ее применение стало популярным из-за своей простоты и эффективности. Функция активации ReLU часто используется во внутренних слоях сверточных нейронных сетей (CNN) и полносвязных нейронных сетей (DNN).

**Параметры обучения**

Для обучения модели были установлены следующие параметры.

1. Loss Function (Функция потерь):

В качестве функции потерь используется бинарная кросс-энтропия. Это часто используется в задачах бинарной классификации.

1. Optimizer (Оптимизатор):

Для оптимизации используется алгоритм оптимизации Adam. Adam обычно эффективен в задачах градиентного спуска.

1. Metrics (Метрики):

* metrics.BinaryAccuracy(): точность, измеряет долю правильно классифицированных экземпляров.
* metrics.AUC(): значение площади под ROC-кривой, что является мерой качества классификации.
* metrics.Precision(): доля положительных предсказаний, которые были верны.
* metrics.Recall(): метрика полноты, измеряющая долю истинных положительных предсказаний относительно всех реальных положительных случаев.

1. Weighted Metrics (Взвешенные метрики):

* weighted\_metrics = ['accuracy']: взвешенная метрика может быть полезной в случае несбалансированных классов, где один класс преобладает над другим.

Кроме того, в настройки процесса обучения были добавлены две callback-функции:

* EarlyStopping – автоматическая остановка обучения модели, когда заданный параметр перестает улучшаться на валидационном наборе данных. Это помогает предотвратить переобучение модели и ускоряет процесс обучения.
* Checkpoint – сохранение лучшей модели в файл.

**Полученные метрики**

Метрики для модели показаны на рисунке 11.

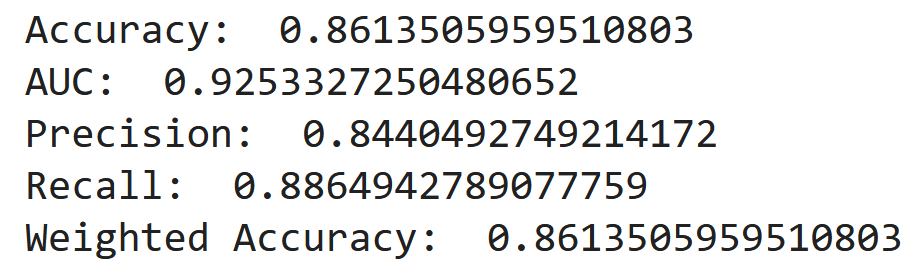


Рисунок 11 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 12).

****

Рисунок 12 — График визуализации обучения

**ROC-кривые**

График ROC-кривой и AUC (рисунок 13).

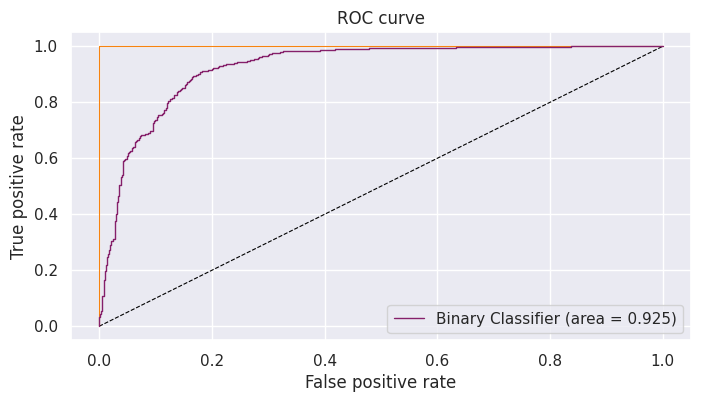
****

Рисунок 13 — График ROC-кривой и AUC

Матрица спутанности (рисунок 14).

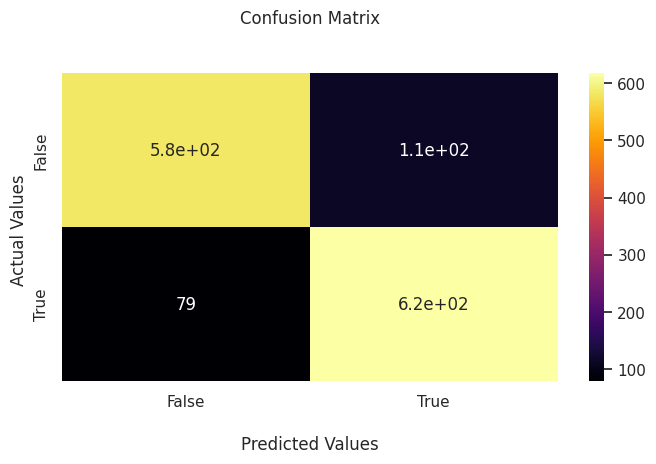


Рисунок 14 — Матрица спутанности

4.1.3.2 Модель №2

**Параметры архитектуры**

Архитектура нейросети включает в себя четыре слоя: с 128 нейронами и активацией ReLU, второй слой с 64 нейронами и активацией ReLU, третий слоем с 32 нейронами и активацией ReLU, и, наконец, выходным слоем с одним нейроном и активацией sigmoid.

**Параметры обучения**

Параметры обучения оставим прежними, за исключением оптимизатора. Выберем вместо Adam алгоритм RMSprop. RMSprop (Root Mean Square Propagation) – это метод оптимизации, который позволяет эффективно обучать нейронные сети, адаптируя скорость обучения для каждого параметра на основе квадратного корня из среднего квадрата предыдущих градиентов.

**Полученные метрики**

Метрики для модели показаны на рисунке 15.

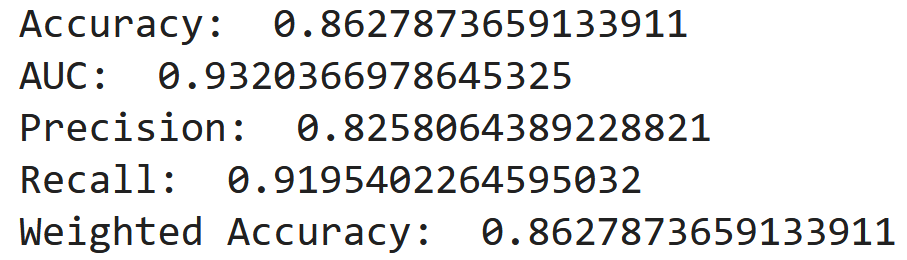


Рисунок 15 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 16).

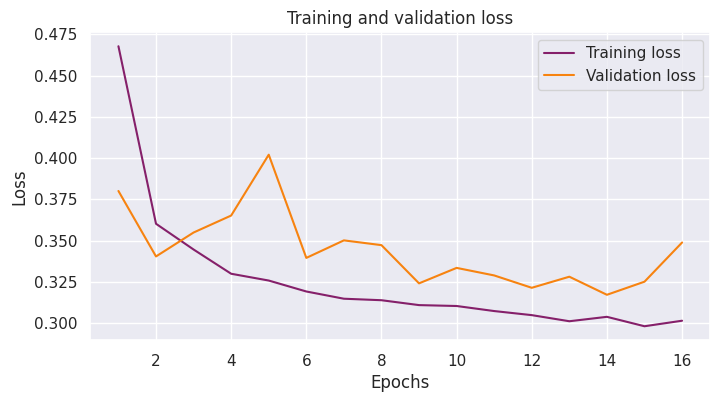
****

Рисунок 16 — График визуализации обучения

**ROC-кривые**

График ROC-кривой и AUC (рисунок 17).

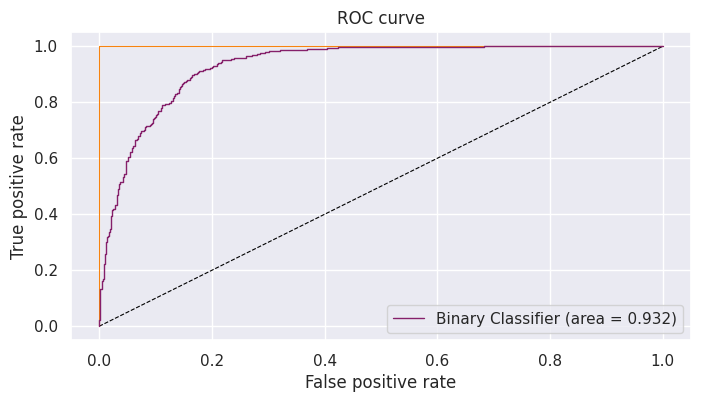


Рисунок 17 — График ROC-кривой и AUC

Матрица спутанности (рисунок 18).

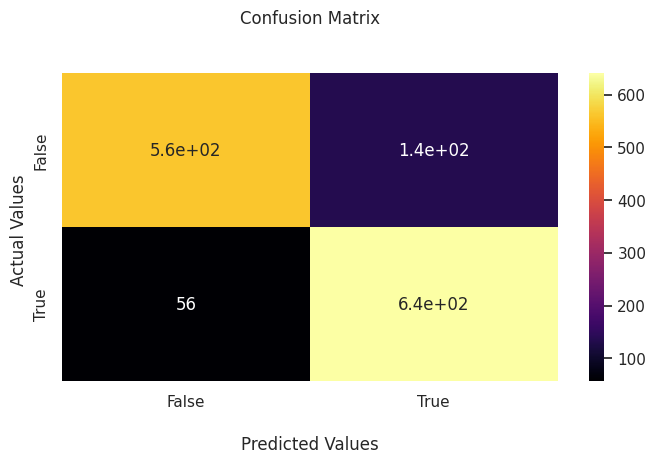


Рисунок 18 — Матрица спутанности

4.1.3.3 Модель №3

**Параметры архитектуры**

Архитектура нейросети включает в себя последовательный бинарный классификатор с пятью слоями: полносвязный слой с 128 нейронами и функцией активации ReLU, слой исключения (Dropout) с коэффициентом отсева 0.05, полносвязный слой с 64 нейронами и функцией активации SELU, cлой исключения (Dropout) с коэффициентом отсева 0.1, полносвязный слой с 32 нейронами и функцией активации ReLU.

Выходной слой с одним нейроном и функцией активации sigmoid.

Функция активации SELU (Scaled Exponential Linear Unit) обычно используется в нейронных сетях, особенно в глубоких сетях. SELU является вариантом функции активации ELU (Exponential Linear Unit), но с автоматическим масштабированием, то есть обладает свойством самонастраиваемости (self-normalizing property), что означает, что активации в сети имеют тенденцию к сохранению статистических характеристик при проходе через слои. Это может способствовать более стабильному обучению глубоких нейронных сетей.

**Параметры обучения**

Выберем в качестве оптимизатора Adam как наиболее универсальный алгоритм, объединяющий идеи двух других популярных оптимизаторов: RMSprop (Root Mean Square Propagation) и метода Momentum.

**Полученные метрики**

Метрики для модели показаны на рисунке 19.

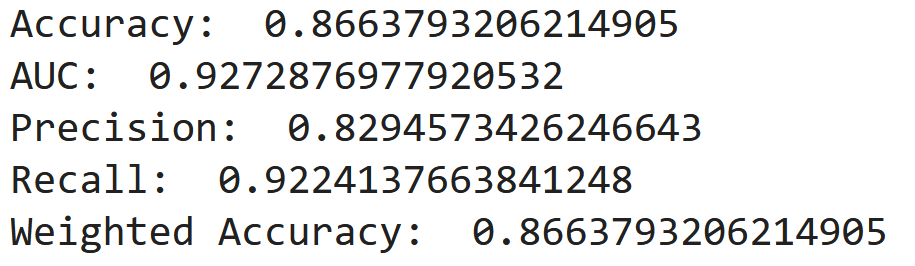


Рисунок 19 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 20).

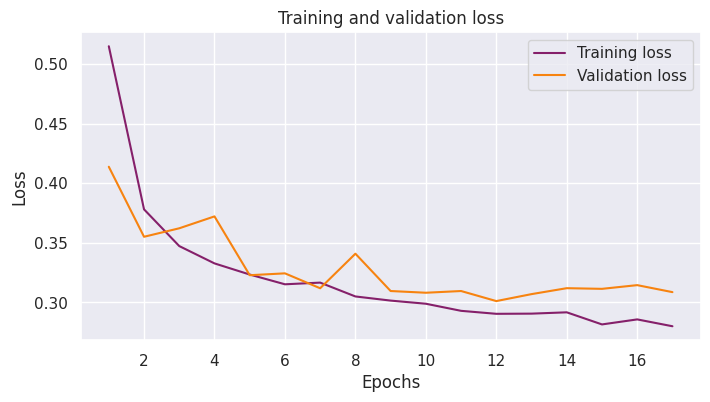
****

Рисунок 20 — График визуализации обучения

**ROC-кривые**

График ROC-кривой и AUC (рисунок 21).

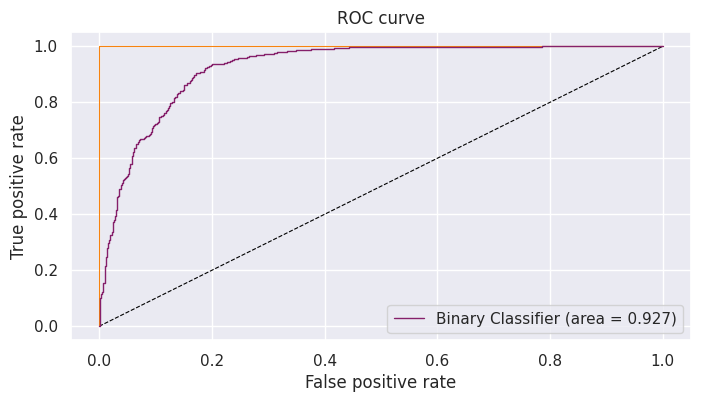


Рисунок 21 — График ROC-кривой и AUC

Матрица спутанности (рисунок 22).

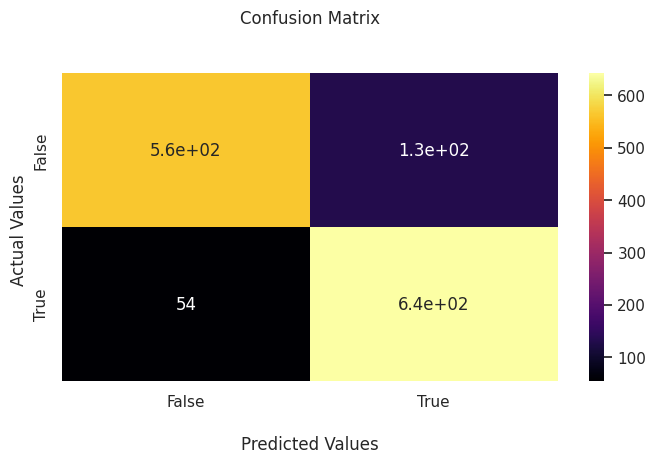


Рисунок 22 — Матрица спутанности

4.1.3.4 Модель №4

**Параметры архитектуры**

Архитектура нейросети включает в себя последовательный бинарный классификатор с пятью слоями: полносвязный слой с 128 нейронами и функцией активации ReLU, слой исключения (Dropout) с коэффициентом отсева 0.05, полносвязный слой с 64 нейронами и функцией активации SELU, cлой исключения (Dropout) с коэффициентом отсева 0.1, полносвязный слой с 32 нейронами и функцией активации ReLU.

Выходной слой с одним нейроном и функцией активации сигмоида.

**Параметры обучения**

К параметрам обучения предыдущей модели добавим уменьшение шага обучения (learning rate) на каждой эпохе обучения. Уменьшение шага обучения может ускорить сходимость алгоритма обучения и сгладить колебания округ локального минимума функции потерь, стабилизировав тем самым процесс обучения.

**Полученные метрики**

Метрики для модели показаны на рисунке 23.

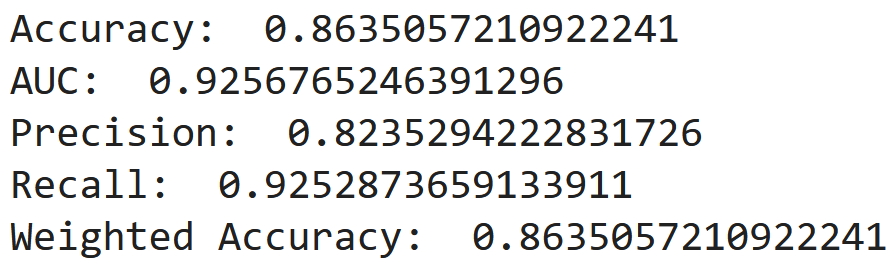


Рисунок 23 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 24).

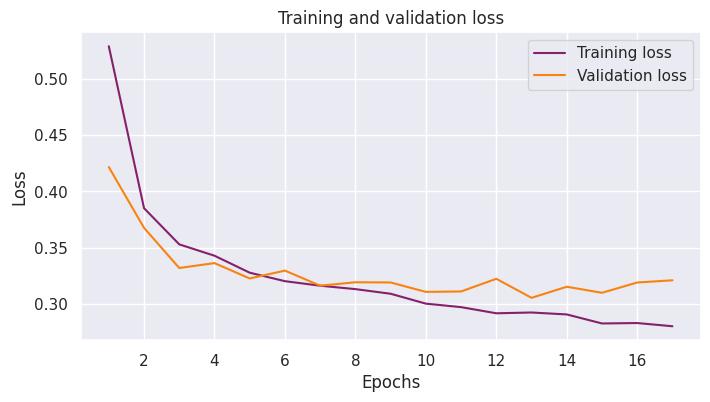
****

Рисунок 24 — График визуализации обучения

**ROC-кривые**

График ROC-кривой и AUC (рисунок 25).

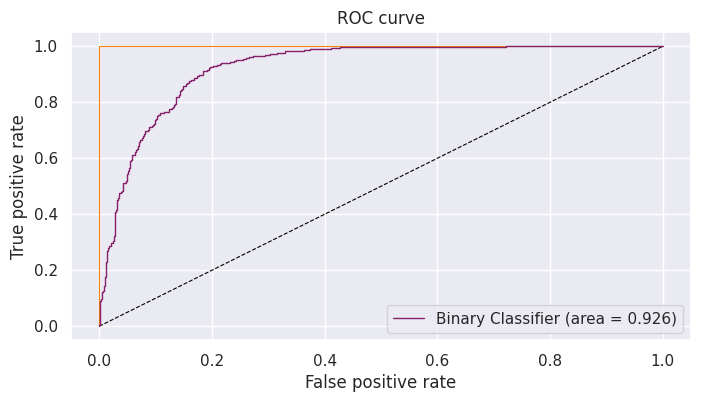


Рисунок 25 — График ROC-кривой и AUC

Матрица спутанности (рисунок 26).

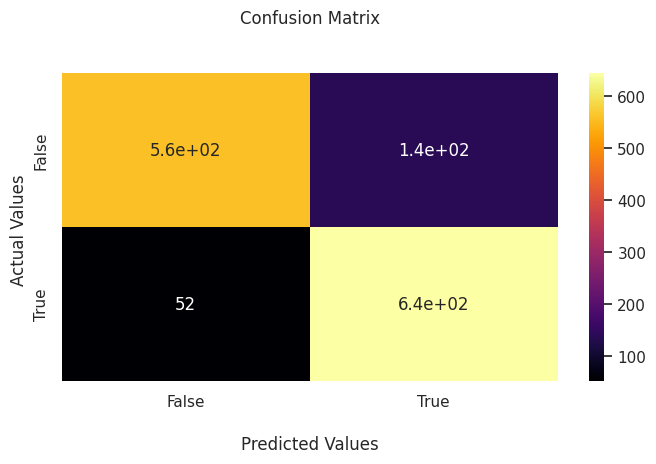


Рисунок 26 — Матрица спутанности

4.1.4 Выводы по задаче бинарной классификации

В целом, все построенные модели достаточно хорошо справились с задачей. Наименее сложную архитектуру при высоком результате обеспечивает Модель №2. График ROC-кривой данной модели также является наиболее гладким.

4.2 Многоклассовый классификатор

4.2.1 Входные данные

Снижение детской смертности отражено в ряде целей Организации Объединенных Наций в области устойчивого развития и является ключевым показателем прогресса человечества.

ООН ожидает, что к 2030 году страны положат конец предотвратимой смертности новорожденных и детей в возрасте до 5 лет, при этом все страны стремятся снизить смертность детей в возрасте до 5 лет по крайней мере до 25 на 1000 живорождений.

Параллельным понятию детской смертности, конечно же, является материнская смертность, на долю которой приходится 295 000 смертей во время беременности и родов и после них (по состоянию на 2017 год). Подавляющее большинство этих смертей (94%) произошло в условиях нехватки ресурсов, и большинство из них можно было предотвратить.

В свете того, что было упомянуто выше, кардиотокограммы (КТГ) являются простым и доступным по стоимости способом оценки здоровья плода, позволяющим медицинским работникам принимать меры для предотвращения детской и материнской смертности. Само оборудование работает, посылая ультразвуковые импульсы и считывая их реакцию, тем самым проливая свет на частоту сердечных сокращений плода, движения плода, сокращения матки и многое другое.

Этот набор данных содержит 2126 записей признаков, извлеченных из результатов кардиотокограмм, которые затем были классифицированы тремя экспертами-акушерами на 3 класса:

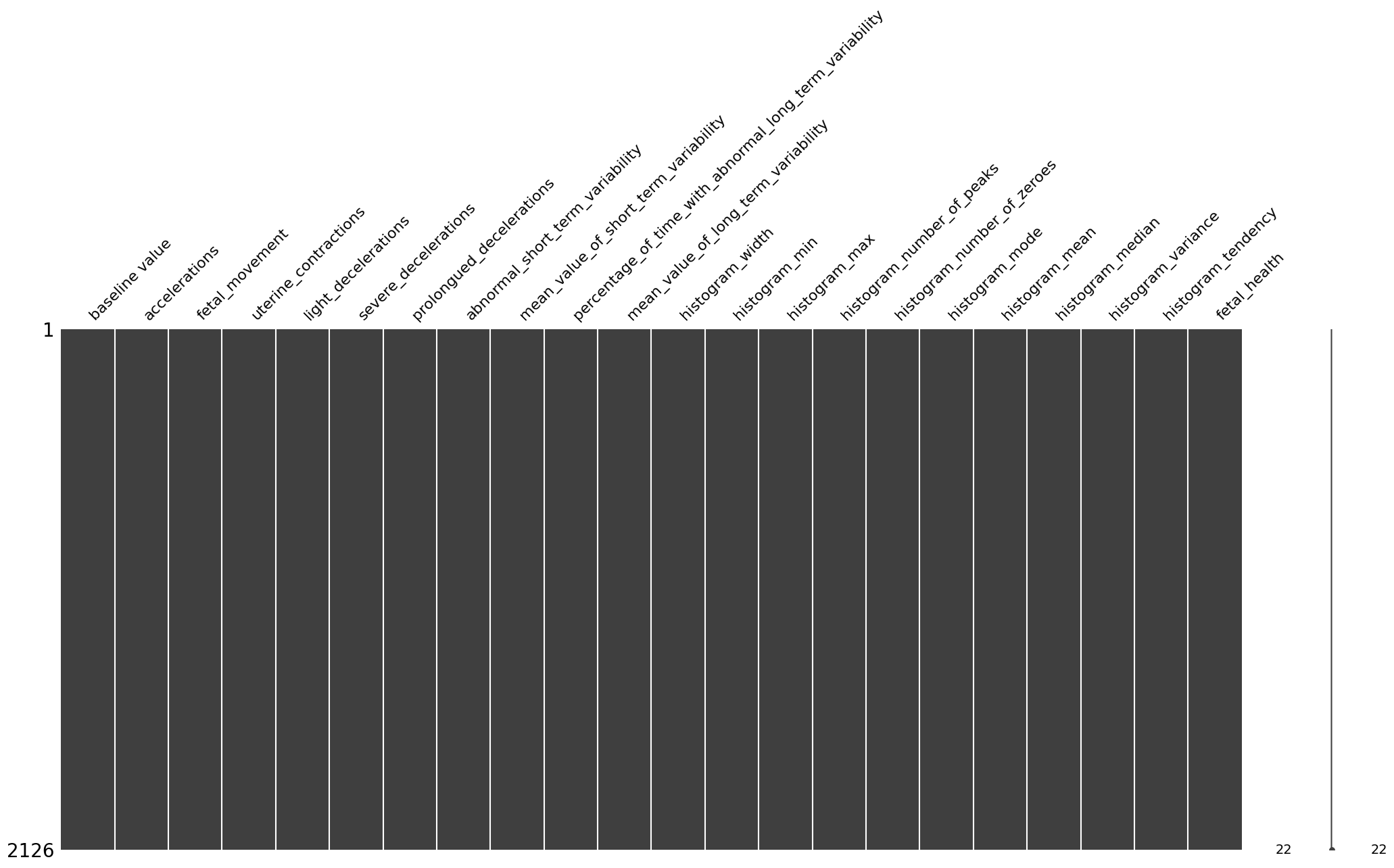
* Normal (Нормальные)
* Suspect (Подозрительные)
* Pathological (Патологические)

Набор данных содержит следующие признаки:

* baseline\_value ‒ исходная частота сердечных сокращений плода;
* accelerations ‒ количество ускорений в секунду;
* fetal\_movement ‒ количество движений плода в секунду;
* uterine\_contractions ‒ количество сокращений матки в секунду;
* light\_decelerations ‒ количество LDS в секунду;
* severe\_deceleration ‒ количество Sds в секунду;
* prolongued\_deceleration ‒ количество PDs в секунду;
* abnormal\_short\_term\_variatability ‒ процент времени с ненормально коротким периодом изменчивости;
* mean\_value\_of\_short\_term\_variability ‒ среднее значение краткосрочной изменчивости;
* percentage\_of\_time\_with\_abnormal\_long\_term\_variability ‒ процент времени с аномальной долгосрочной изменчивостью;
* mean\_value\_of\_long\_term\_variability ‒ среднее значение длительной вариабельности;
* histogram\_width ‒ ширина гистограммы с использованием всех значений из записи;
* histogram\_min ‒ минимальное значение гистограммы;
* histogram\_max ‒ максимальное значение гистограммы;
* histogram\_number\_of\_peaks ‒ количество пиков в гистограмме;
* histogram\_number\_of\_zeroes ‒ количество нулей в гистограмме;
* histogram\_mode ‒ режим гистограммы;
* histogram\_mean ‒ среднее значение гистограммы;
* histogram\_median ‒ медиана гистограммы;
* histogram\_variance ‒ дисперсия гистограммы;
* histogram\_tendency ‒ тенденция гистограммы;
* fetal\_health ‒ здоровье плода.
  + 1. Предобработка и разведывание данных

**Проверка пропущенных значений**

Используя библиотеку missingno, убеждаемся, что в датасете нет пропусков (рисунок 27).

Рисунок 27 — Пропущенные значения

**Проверка на сбалансированность**

Распределение данных по классам целевой переменной (рисунок 28). По диаграмме можно судить о несбалансированности датасета.

****

Рисунок 28 — Распределение данных по классам целевой переменной

При решении данной задачи для балансирования данных был использован метод oversampling.

**Матрица корреляции**

На рисунке 29 представлена матрица корреляции.

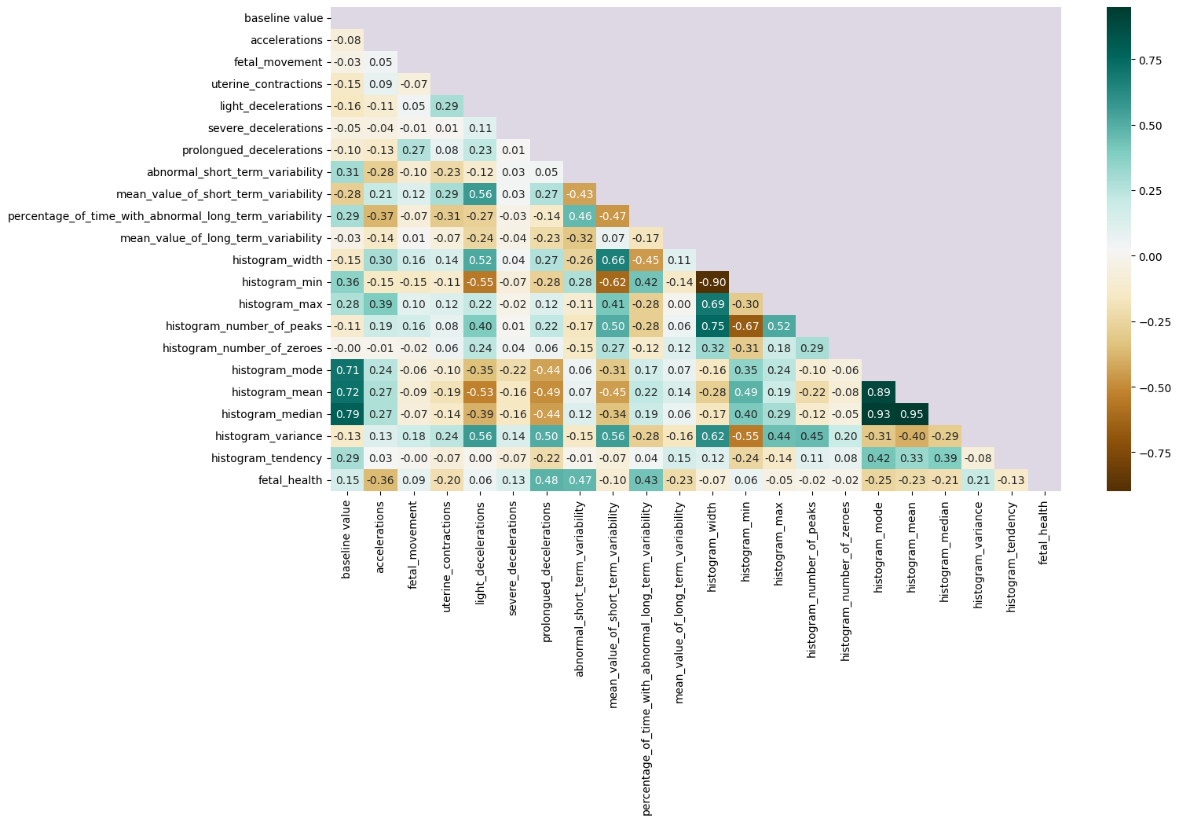
****

Рисунок 29 — Матрица корреляции

По матрице корреляции можно заключить, что признаки accelerations, prolongued\_decelerations, abnormal\_short\_term\_variability, percentage\_of\_time\_with\_abnormal\_long\_term\_variability and mean\_value\_of\_long\_term\_variability наиболее сильно связаны с целевой переменной fetal\_health.

**Анализ зависимостей между признаками**

**Зависимость accelerations (ускорения) и движения плода (fetal\_movement)**

Зависимость переменных представлена на рисунках 30 и 31.

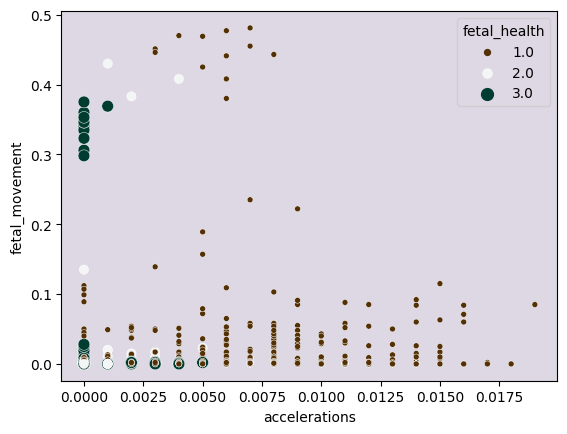
****

Рисунок 30 — fetal\_movement и accelerations

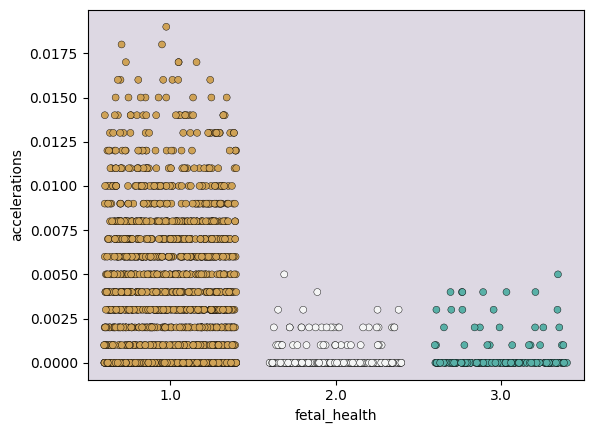
****

Рисунок 31 — accelerations

Патологические и подозрительные значения располагаются близко к нулю по оси абсцисс. Чем меньше значение accelerations, тем более вероятно, что значения подозрительные или патологические.

**Зависимость prolonged decelerations (длительное замедление) и fetal movement (движение плода)**

Зависимость переменных представлена на рисунках 32, 33 и 34.

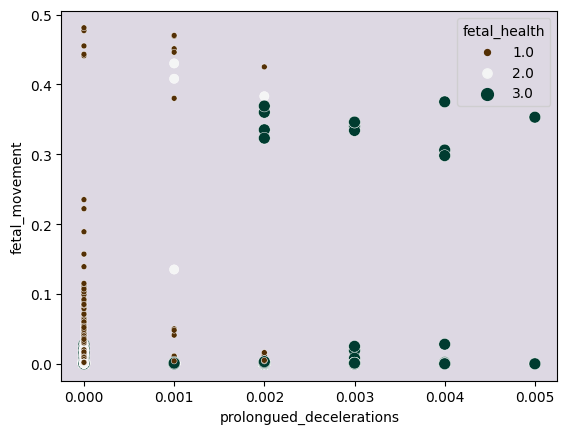
****

Рисунок 32 — prolonged decelerations и fetal movement

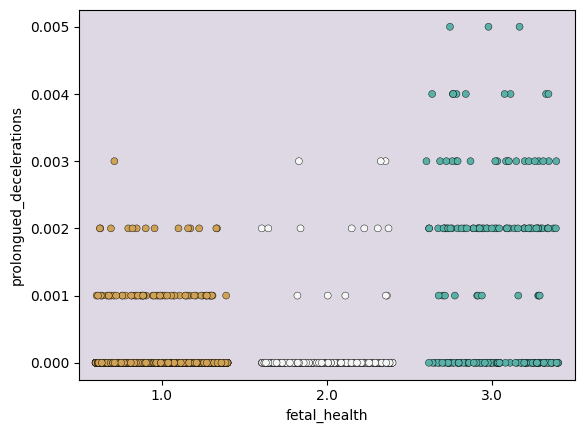
****

Рисунок 33 — prolonged decelerations

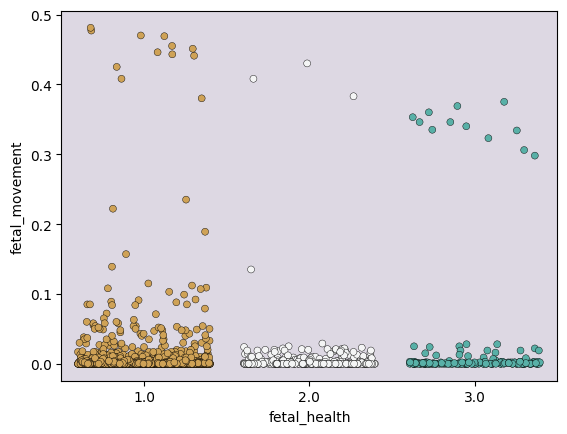
****

Рисунок 34 — fetal movement

Соответственно, в случае признака prolonged decelerations имеется обратная тенденция.

**Зависимость abnormal short term variability (аномальная кратковременная изменчивость) и fetal movement (движение плода)**

Зависимость переменных представлена на рисунках 35, 36.

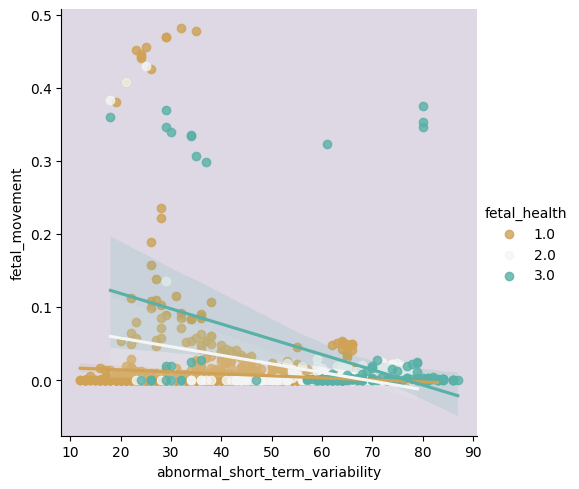
****

Рисунок 35 — abnormal short term variability и fetal movement

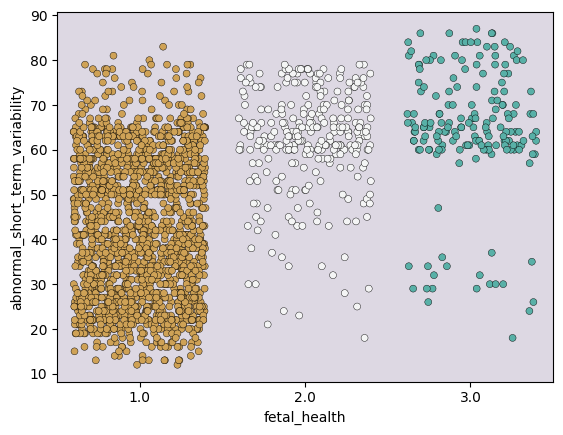
****

Рисунок 36 — abnormal short term variability

**Зависимость mean value of long term variability (среднее значение долгосрочной изменчивости) от fetal movement (движение плода)**

Зависимость переменных представлена на рисунках 37, 38.

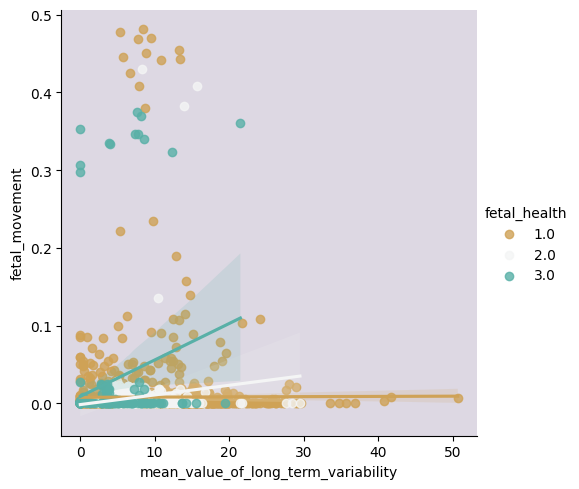
****

Рисунок 37 — mean value of long term variability и fetal movement

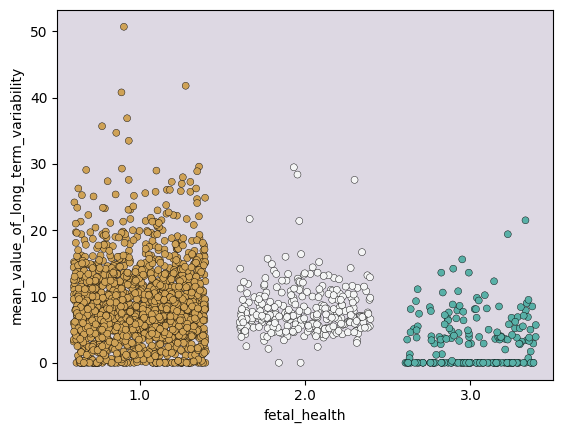
****

Рисунок 38 — mean value of long term variability

Как мы видим из графиков, изменение каждого из рассмотренных выше признаков определенным образом влияет на целевую переменную.

**Проверка выбросов**

Боксплоты для каждого признака представлены на рисунке 39.

|  |  |  |
| --- | --- | --- |
| **C:\Users\User\Downloads\22.png** | **C:\Users\User\Downloads\20.png** | **C:\Users\User\Downloads\19.png** |
| **C:\Users\User\Downloads\18.png** | **C:\Users\User\Downloads\17.png** | **C:\Users\User\Downloads\16.png** |
| **C:\Users\User\Downloads\15.png** | **C:\Users\User\Downloads\14.png** | **C:\Users\User\Downloads\13.png** |
| **C:\Users\User\Downloads\12.png** | **C:\Users\User\Downloads\11.png** | **C:\Users\User\Downloads\10.png** |
| **C:\Users\User\Downloads\9.png** | **C:\Users\User\Downloads\8.png** | **C:\Users\User\Downloads\7.png** |
| **C:\Users\User\Downloads\6.png** | **C:\Users\User\Downloads\5.png** | **C:\Users\User\Downloads\4.png** |
| **C:\Users\User\Downloads\3.png** | **C:\Users\User\Downloads\2.png** | **C:\Users\User\Downloads\1.png** |

Рисунок 39 — Выбросы

Поскольку мы рассматриваем задачу классификации разделения данных на нормальные, подозрительные и патологические, удаление выбросов может значительно снизить качество модели. Метки были присвоены экспертами, поэтому удаление выбросов в данной задаче, вероятно, скажется негативно на процессе обучения.

**Использованные признаки**

Для анализа данных будем использовать все признаки из набора.

4.2.3 Построенные нейросетевые модели

4.2.3.1 Модель №1

**Параметры архитектуры**

Архитектура нейронной сети представляет собой последовательность слоев:

* Первый слой является входным слоем, который принимает данные и передает их в следующий слой.
* Второй слой является полносвязным слоем с 64 нейронами и функцией активации ReLU.
* Третий слой также является полносвязным слоем с 32 нейронами и функцией активации ReLU.
* Четвертый слой является выходным слоем с количеством нейронов, соответствующим общему количеству классов, и функцией активации softmax.

Таким образом, данная нейронная сеть имеет два скрытых слоя с нейронами, соответственно, 64 и 32, и выходной слой, который определяет вероятности принадлежности к каждому из классов.

Функция активации softmax обычно используется в задачах классификации, где необходимо предсказать вероятности каждого класса. Например, в задачах многоклассовой классификации, где объекты данных относятся к одному из нескольких классов, функция softmax может быть использована для преобразования выходов нейронной сети в вероятности принадлежности каждому классу.

**Параметры обучения**

Используются следующие параметры обучения нейросети:

* Оптимизатор: "adam".
* Функция потерь: "categorical\_crossentropy". Функция потерь, также известная как категориальная перекрестная энтропия, используется для оценки расхождения между предсказаниями модели и истинными значениями целевой переменной. Она часто применяется в задачах классификации с несколькими классами.
* Метрики: metrics.CategoricalAccuracy(), metrics.AUC(), metrics.Precision(), metrics.Recall().
* Взвешенные метрики: ['accuracy'].

**Полученные метрики**

Метрики для модели представлены на рисунке 40.

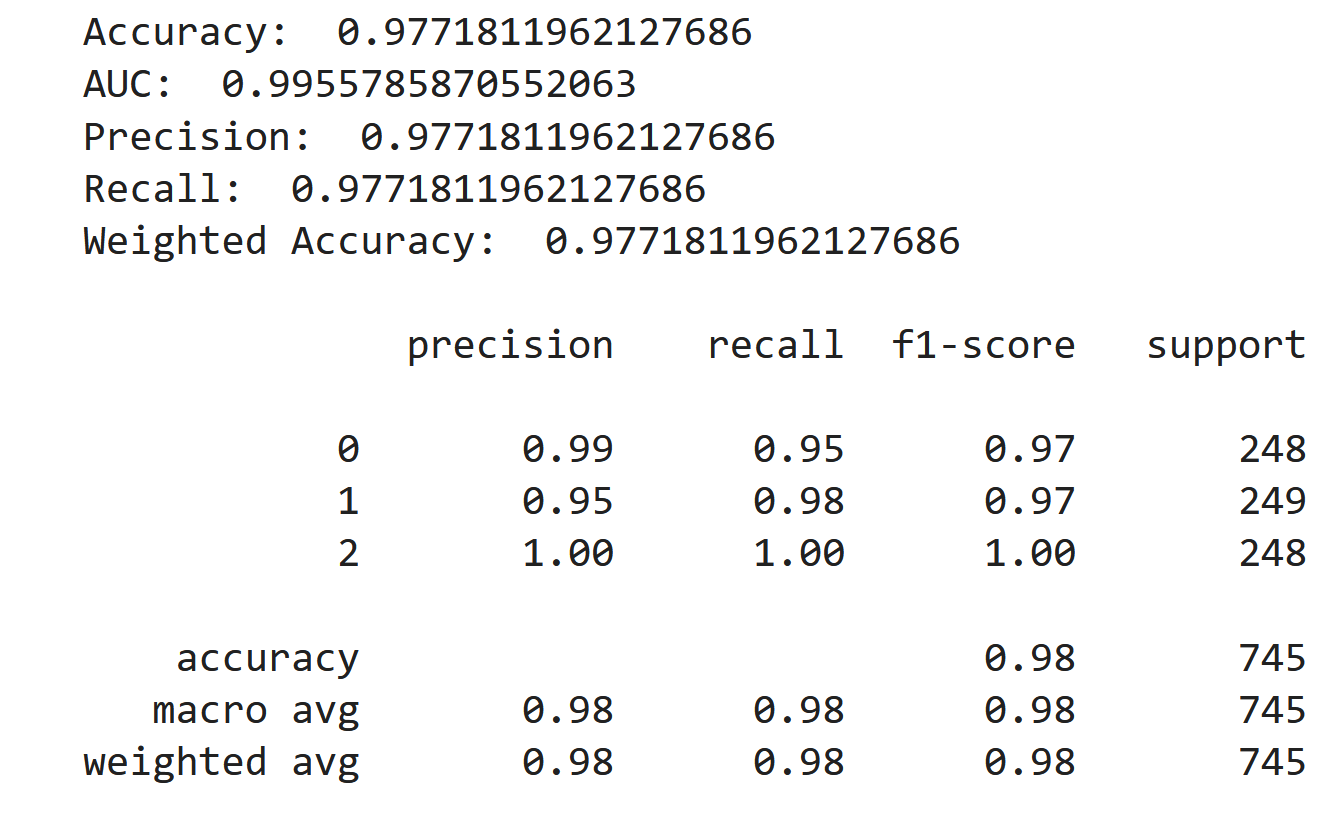
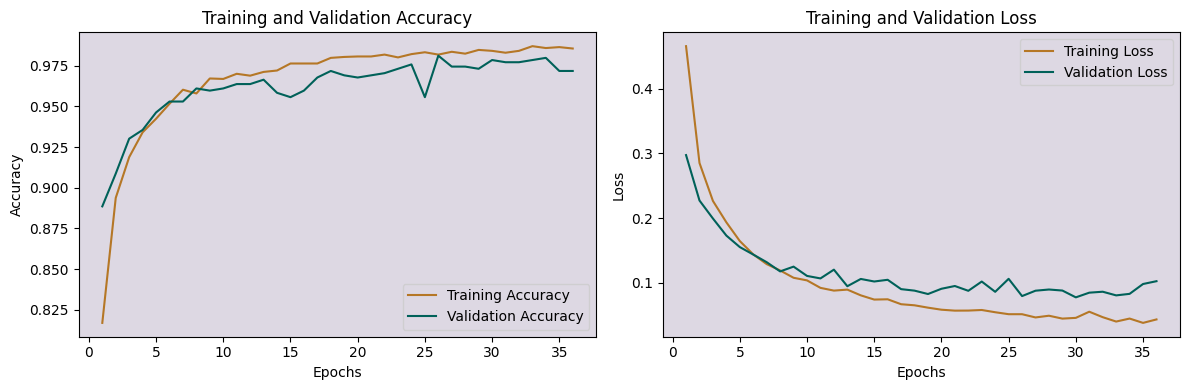


Рисунок 40 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 41).

****Рисунок 41 — График визуализации обучения

Оценка предсказаний (рисунок 42).

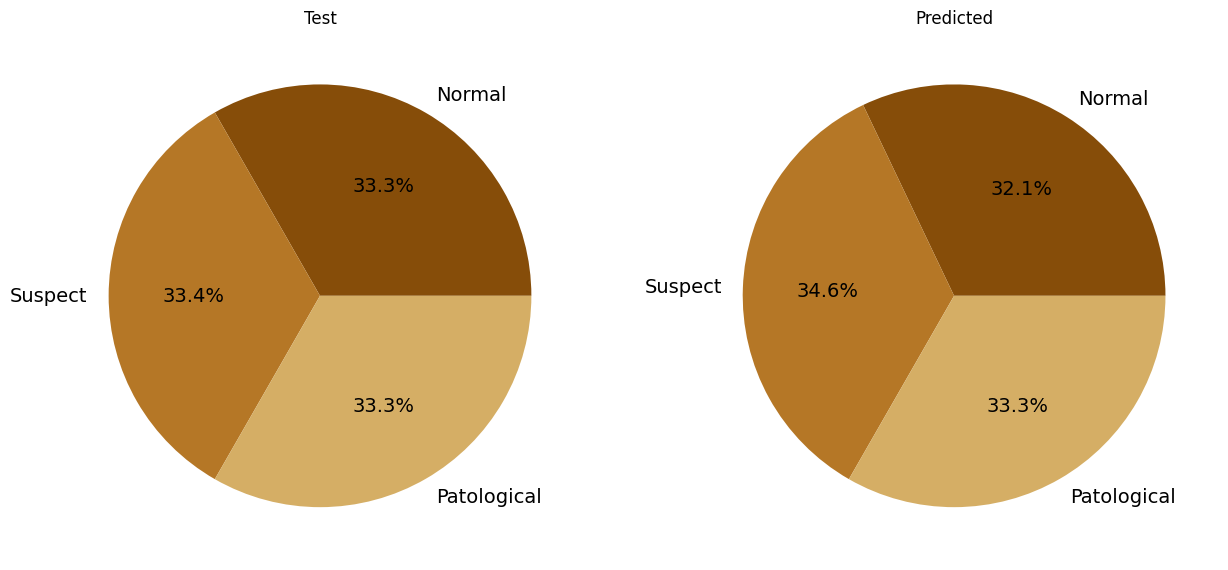
****

Рисунок 42 — Оценка предсказаний

**ROC-кривые**

График ROC-кривой и AUC (рисунок 43).

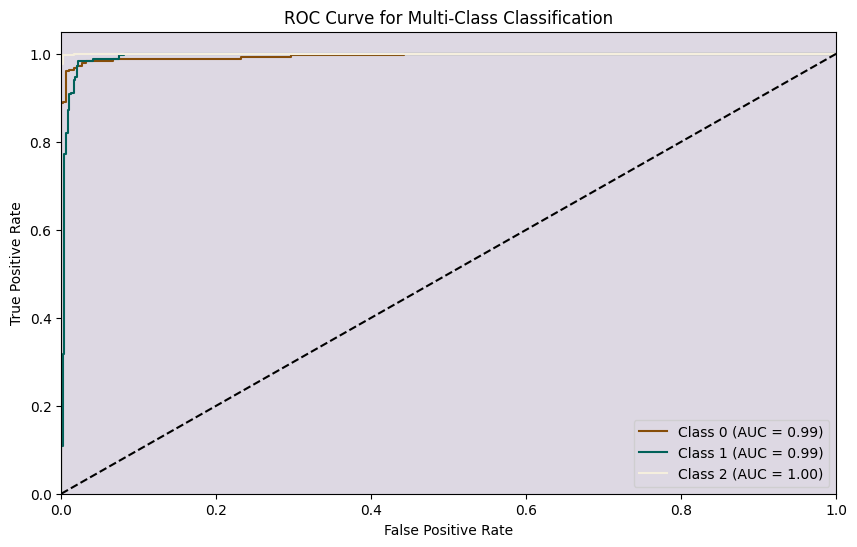
****

Рисунок 43 — График ROC-кривой и AUC

Матрица спутанности (рисунок 44).

|  |  |  |
| --- | --- | --- |
| **C:\Users\User\Downloads\111.png** | **C:\Users\User\Downloads\222.png** | **C:\Users\User\Downloads\333.png** |

Рисунок 44 — Матрица спутанности

4.2.3.2 Модель №2

**Параметры архитектуры**

Данная модель нейронной сети имеет два слоя:

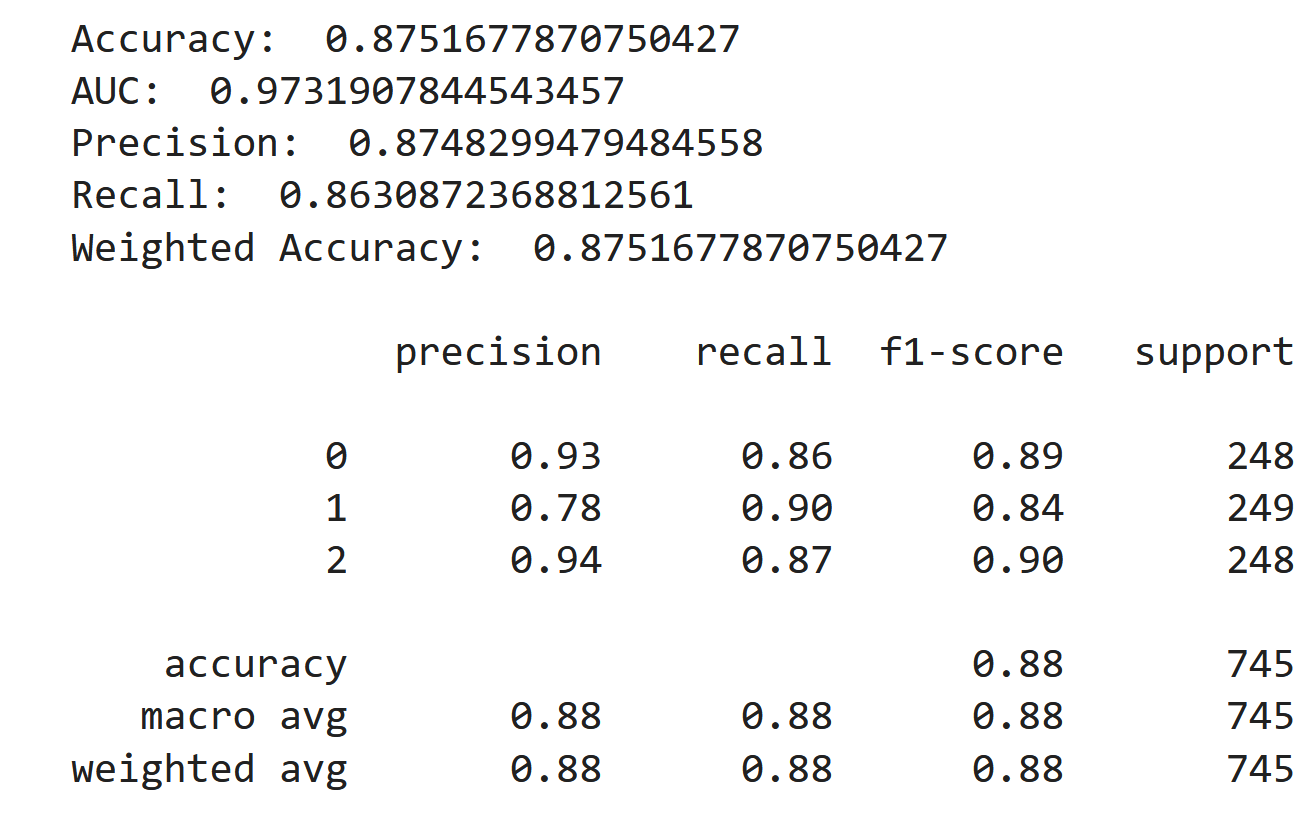
* Первый слой - полносвязный слой с размерностью выхода 3, активационной функцией ReLU.
* Второй слой - полносвязный слой с размерностью выхода, равной общему числу классов в задаче, и активационной функцией softmax.

**Параметры обучения**

Отличием от параметров предыдущей модели является использование оптимизатора RMSprop.

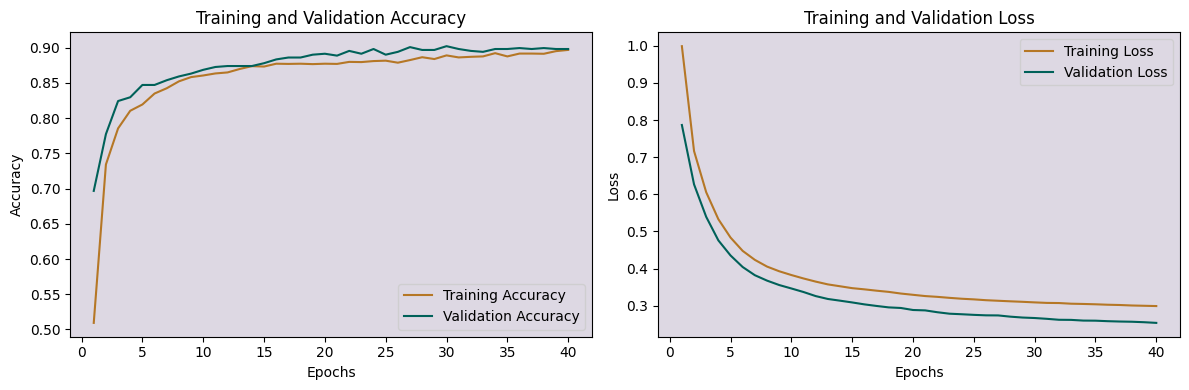
**Полученные метрики**

Метрики для модели представлены на рисунке 45.

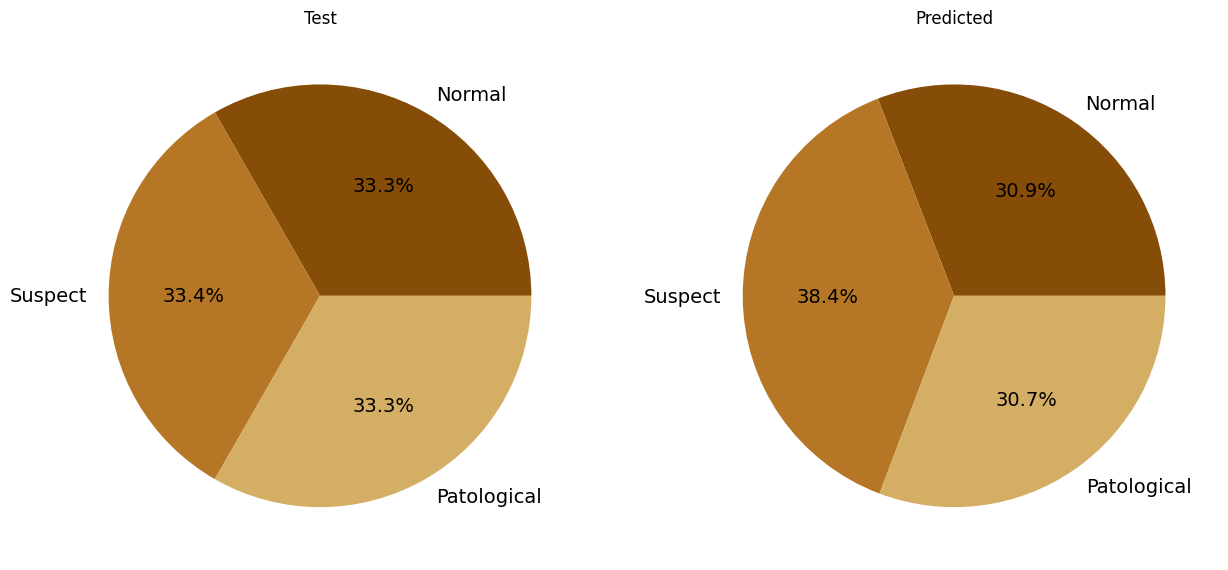
Рисунок 45 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 46).

****Рисунок 46 — График визуализации обучения

Оценка предсказаний (рисунок 47).

****Рисунок 47 — Оценка предсказаний

**ROC-кривые**

График ROC-кривой и AUC (рисунок 48).

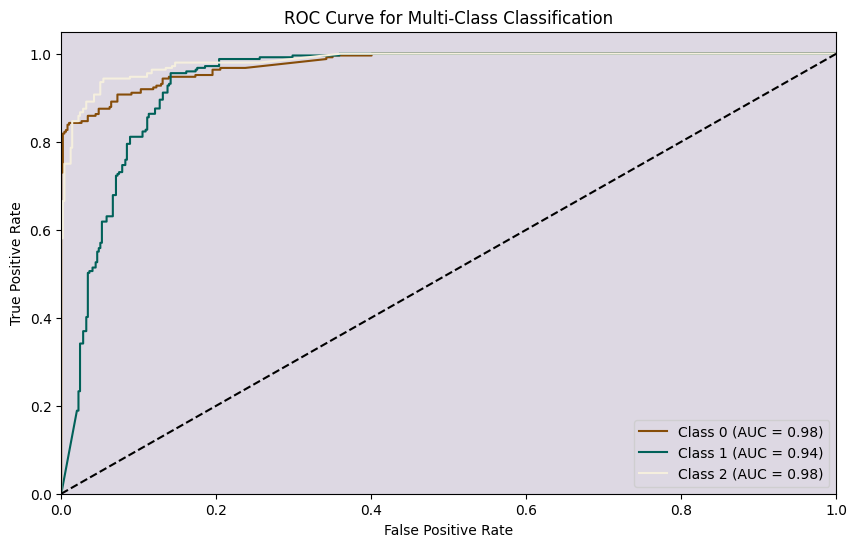
****

Рисунок 48 — График ROC-кривой и AUC

Матрица спутанности (рисунок 49).

|  |  |  |
| --- | --- | --- |
| **C:\Users\User\Downloads\222.png** | **C:\Users\User\Downloads\111.png** | **C:\Users\User\Downloads\333.png** |

Рисунок 49 — Матрица спутанности

4.2.3.3 Модель №3

**Параметры архитектуры**

Возьмем за основу третьей модели модель №1 и изменим функцию активации выходного слоя на сигмоиду.

Если задача подразумевает собой классификацию, при которой метки являются взаимоисключающими, необходимо использовать softmax. softmax выдаст распределение вероятностей, что означает, что все выходные значения будут в сумме равны 1.

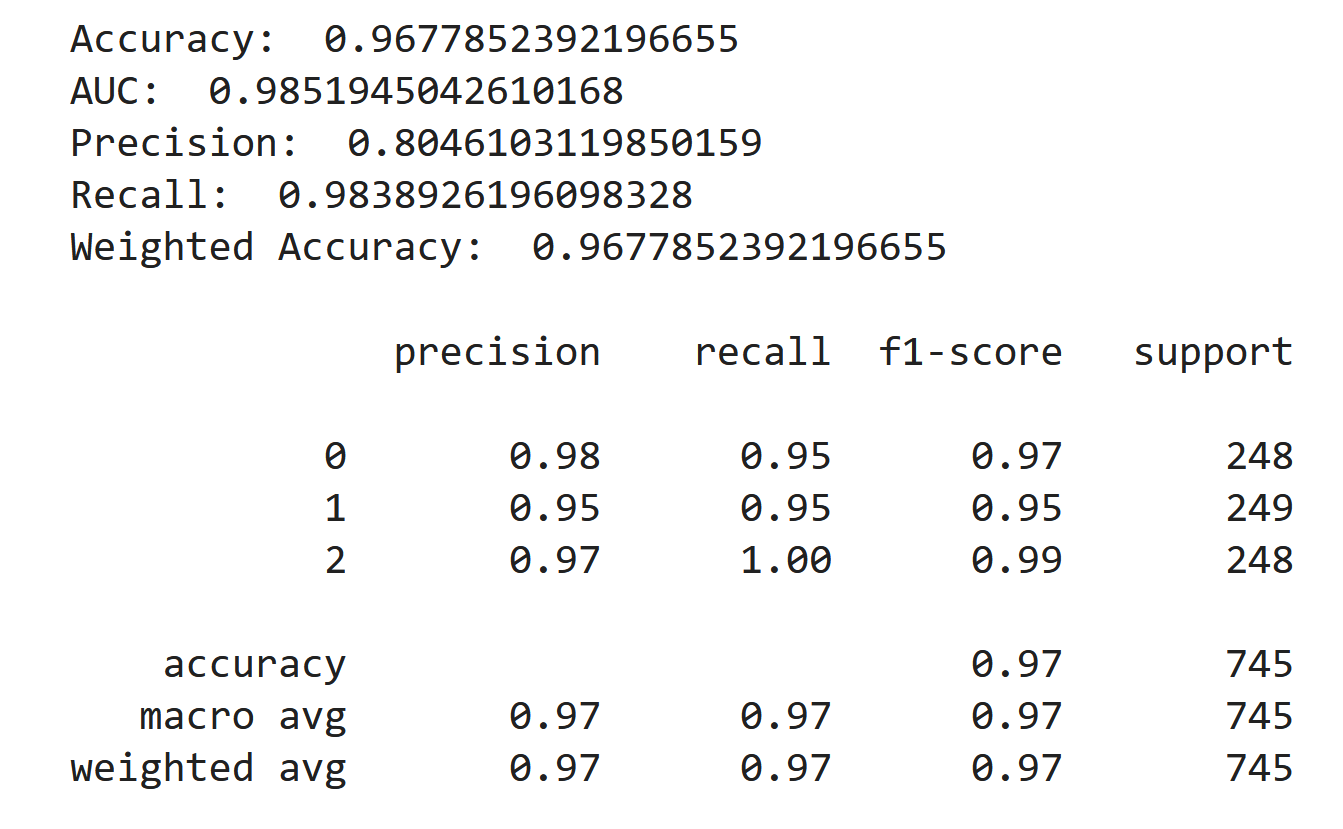
Если входные данные задачи классификации содержат несколько меток для входных данных, то есть классы не являются взаимоисключающими, можно использовать сигмоиду в качестве функции активации. В первом случае вы должны выбрать выходную запись с максимальным значением в качестве выходных данных. В этом случае выход каждого нейрона будет в диапазоне от 0 до 1. Затем можно взять максимальное значение и присвоить объекту класс, соответствующий этому самому значению.

**Параметры обучения**

Основные параметры остались прежними, в качестве оптимизатора используется Adam.

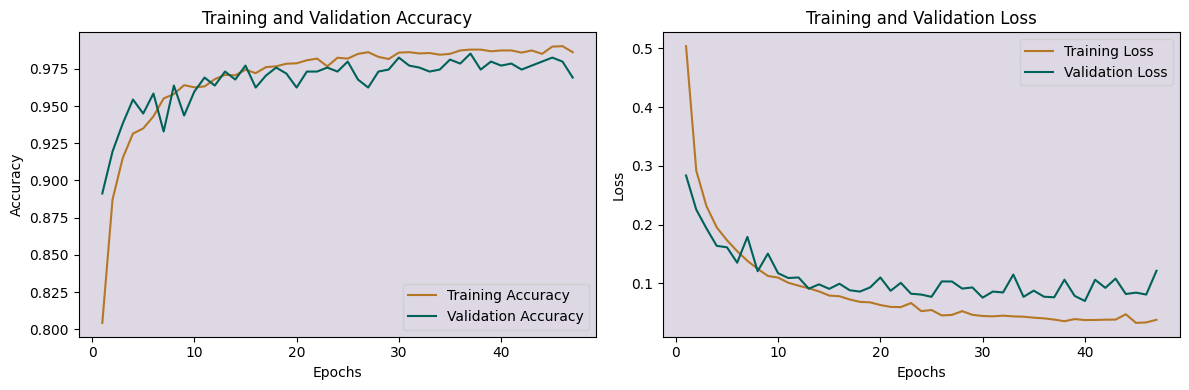
**Полученные метрики**

Метрики для модели представлены на рисунке 50.

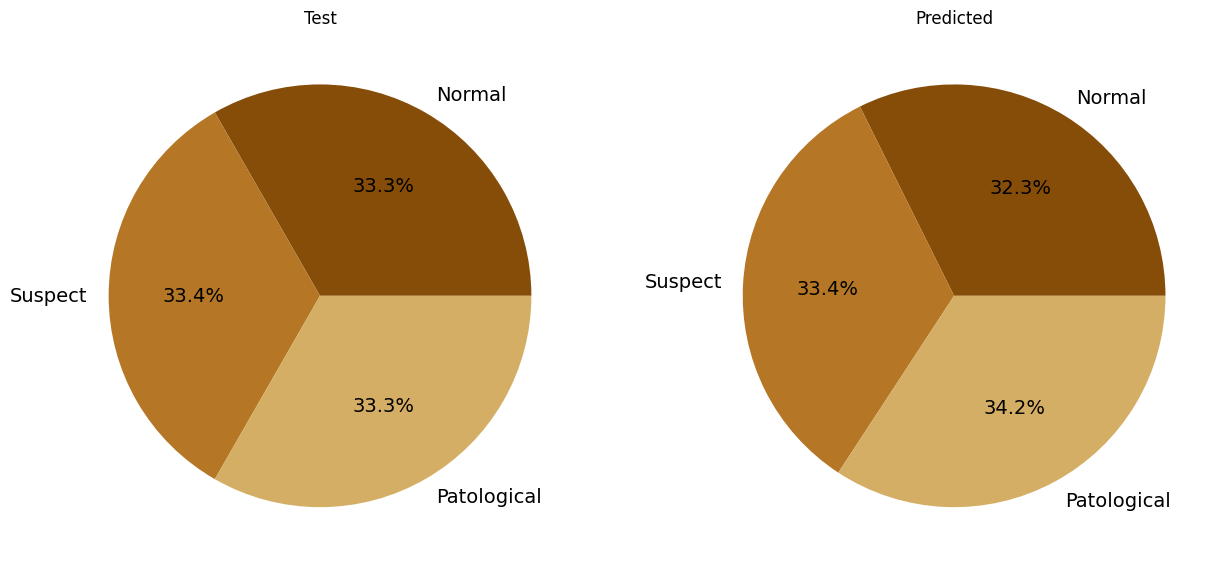
Рисунок 50 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 51).

****Рисунок 51 — График визуализации обучения

Оценка предсказаний (рисунок 52).

****Рисунок 52 — Оценка предсказаний

**ROC-кривые**

График ROC-кривой и AUC (рисунок 53).

****

Рисунок 53 — График ROC-кривой и AUC

Матрица спутанности (рисунок 54).

|  |  |  |
| --- | --- | --- |
| **C:\Users\User\Downloads\111.png** | **C:\Users\User\Downloads\222.png** | **C:\Users\User\Downloads\333.png** |

Рисунок 54 — Матрица спутанности

4.2.3.4 Модель №4

**Параметры архитектуры**

Нейронная сеть, представленная выше, имеет следующую архитектуру:

* Первый слой - полносвязный слой с 128 нейронами и функцией активации ReLU.
* Второй слой - полносвязный слой с 64 нейронами и функцией активации SELU.
* Третий слой - полносвязный слой с 32 нейронами и функцией активации ReLU.
* Последний слой - полносвязный слой с количеством нейронов, соответствующем общему количеству классов, и функцией активации softmax. Этот слой используется для получения вероятностных предсказаний для каждого класса.

**Параметры обучения**

Основные параметры остались прежними, в качестве оптимизатора используется SGD. SGD (Stochastic Gradient Descent) в keras - это оптимизатор, который используется для обучения нейронных сетей. Оптимизатор SGD обновляет веса модели, минимизируя функцию потерь, с использованием градиентного спуска.

**Полученные метрики**

Метрики для модели представлены на рисунке 55.

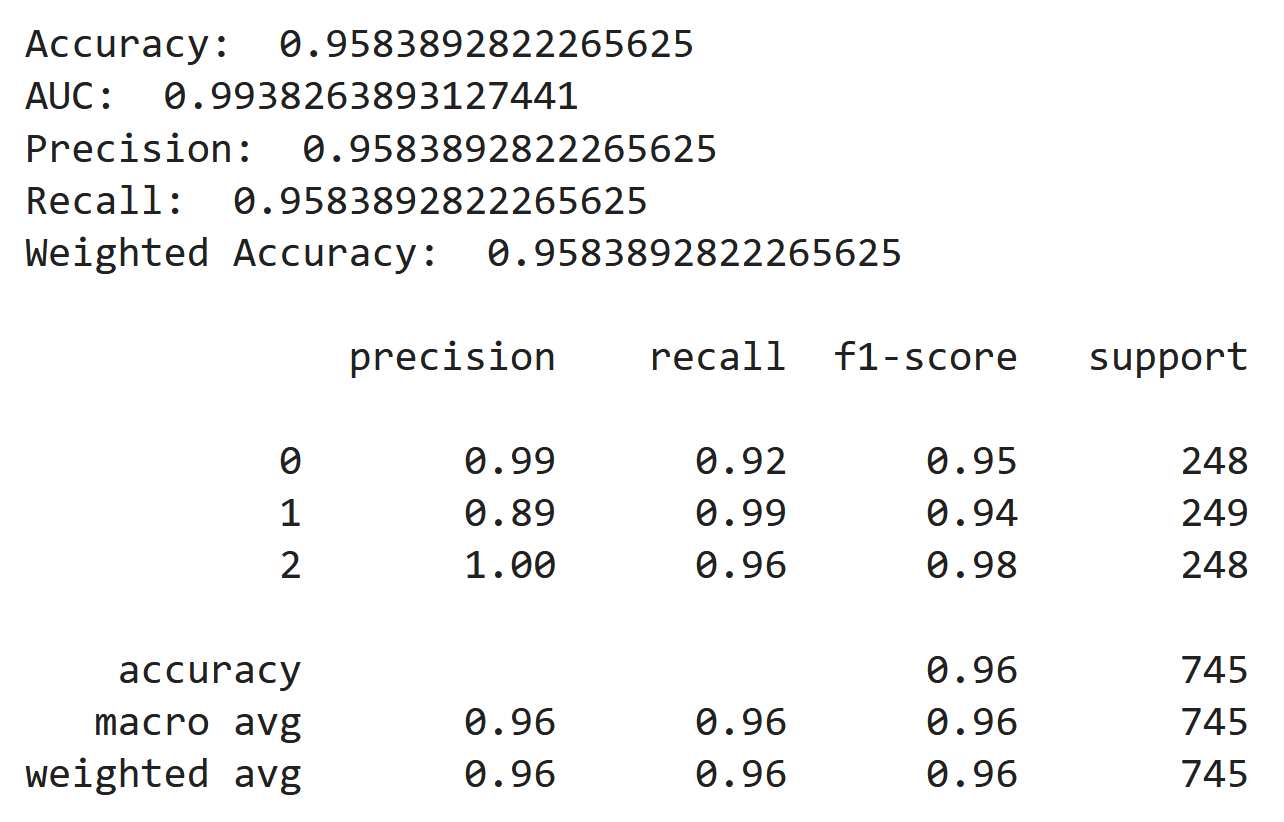
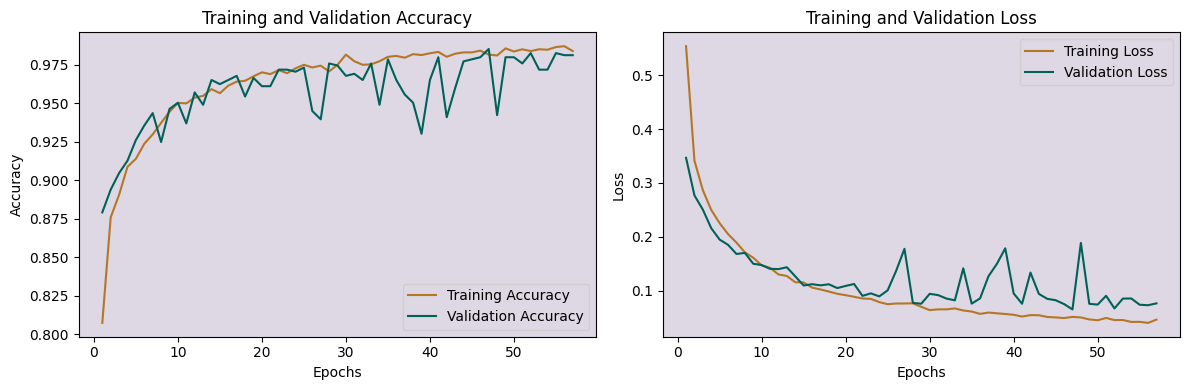


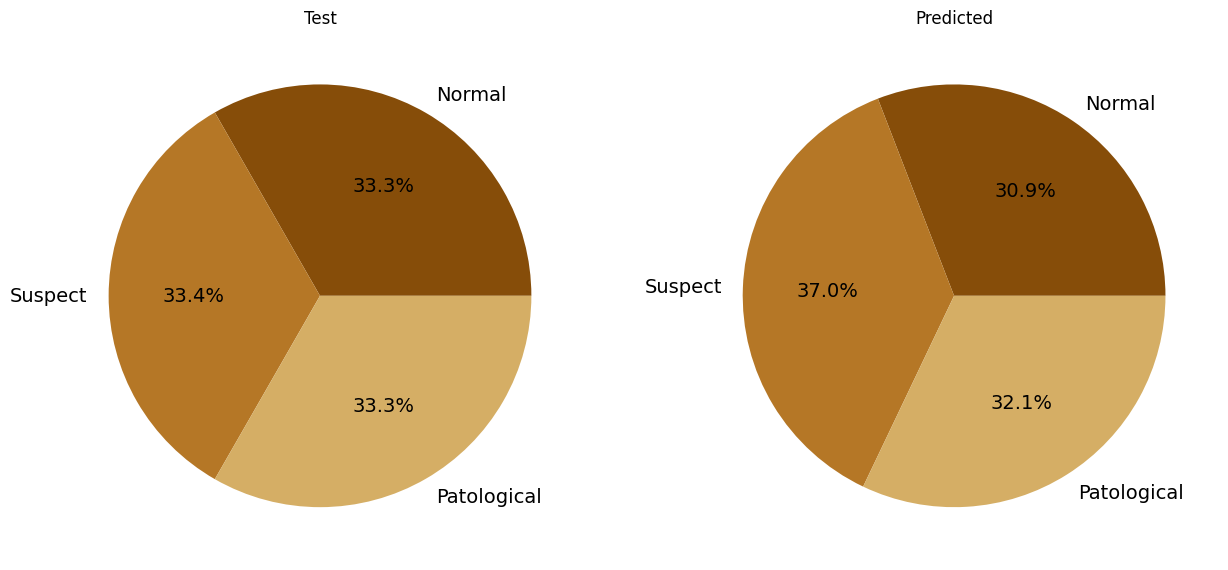
Рисунок 55 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 56).

****Рисунок 56 — График визуализации обучения

Оценка предсказаний (рисунок 57).

****Рисунок 57 — Оценка предсказаний

**ROC-кривые**

График ROC-кривой и AUC (рисунок 58).

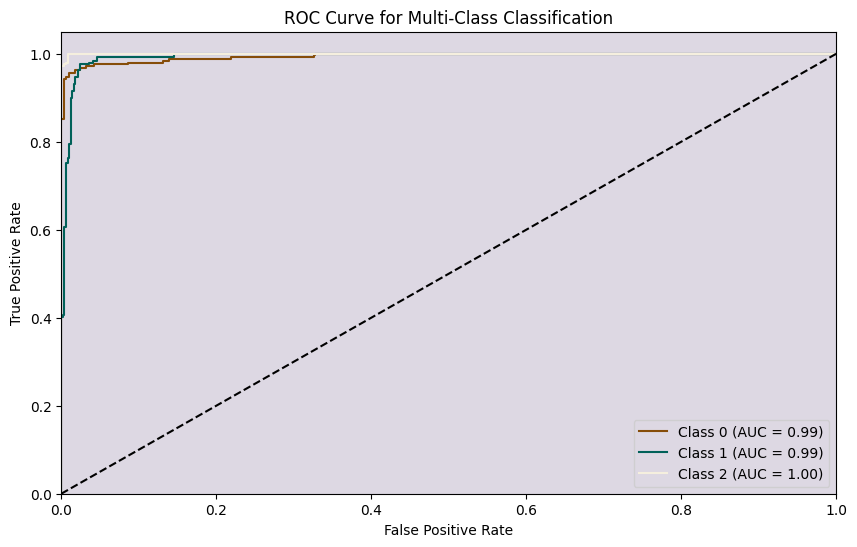
****

Рисунок 58 — График ROC-кривой и AUC

Матрица спутанности (рисунок 59).

|  |  |  |
| --- | --- | --- |
| C:\Users\User\Downloads\111.png | C:\Users\User\Downloads\222.png | C:\Users\User\Downloads\333.png |

Рисунок 59 — Матрица спутанности

4.2.4 Выводы по задаче многоклассовой классификации

Наилучшие результаты продемонстрировала модель №1, средняя по количеству слоёв среди рассмотренных. Любопытен факт, что нейронная сеть с двумя слоями, в которых 3 и 1 нейронов соответственно, также показала хорошие результаты. Возможно, подобного удалось добиться отчасти благодаря предварительной обработке данных, и отчасти так произошло ввиду хорошей разделимости классов (то есть из-за специфики датасета).

4.3 Нейросетевой регрессор

4.3.1 Входные данные

В данной работе был использован набор данных DS\_2019\_public.csv, содержащий информацию о потреблении ресурсов домохозяйством. Описание атрибутов содержится в файле recs2009\_public\_codebook.xlsx. Всего датасет содержит 121 признак.

В качестве целевой переменной была выбрана TOTALBTU - общее потребление (в тысячах BTU)

4.3.2 Предобработка и разведывание данных

**Проверка пропущенных значений**

Используя библиотеку missingno, убеждаемся, что в датасете нет пропусков (рисунок 60).

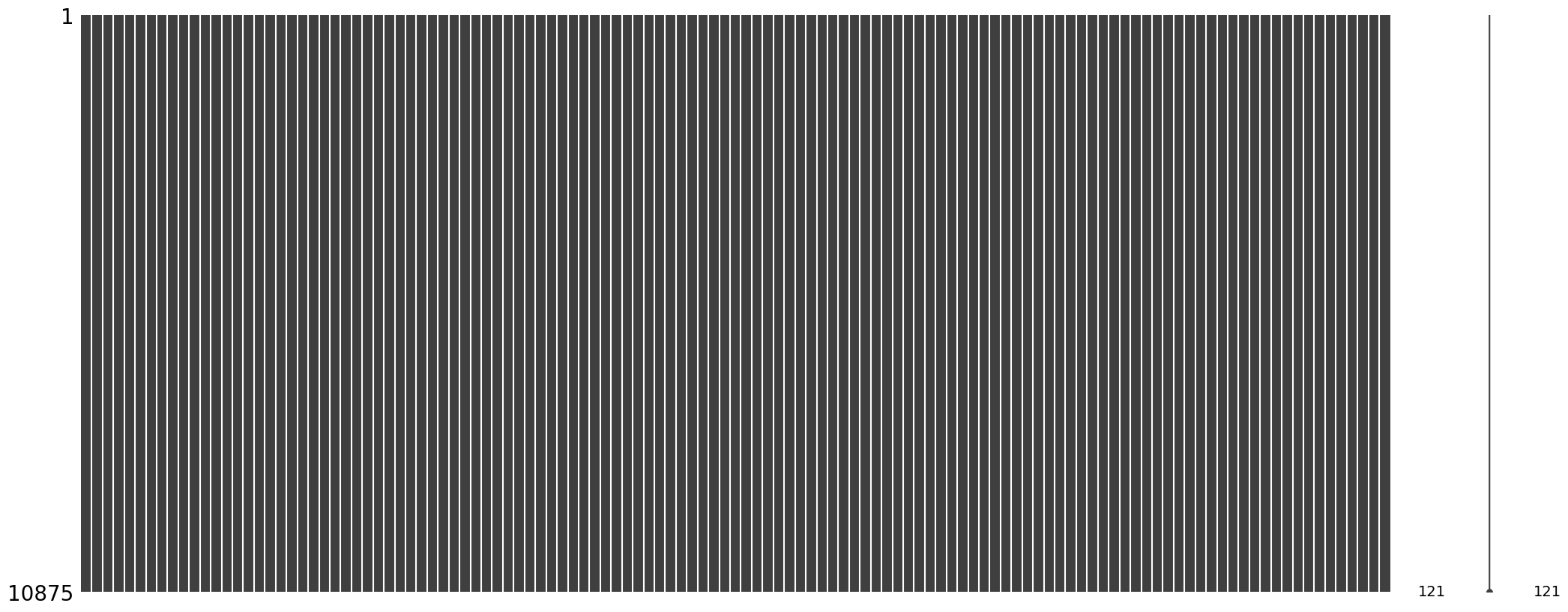


Рисунок 60 — Пропуски

**Матрица корреляции**

Визуально видно, что существует достаточно сильная корреляция между некоторыми признаками (рисунок 61).

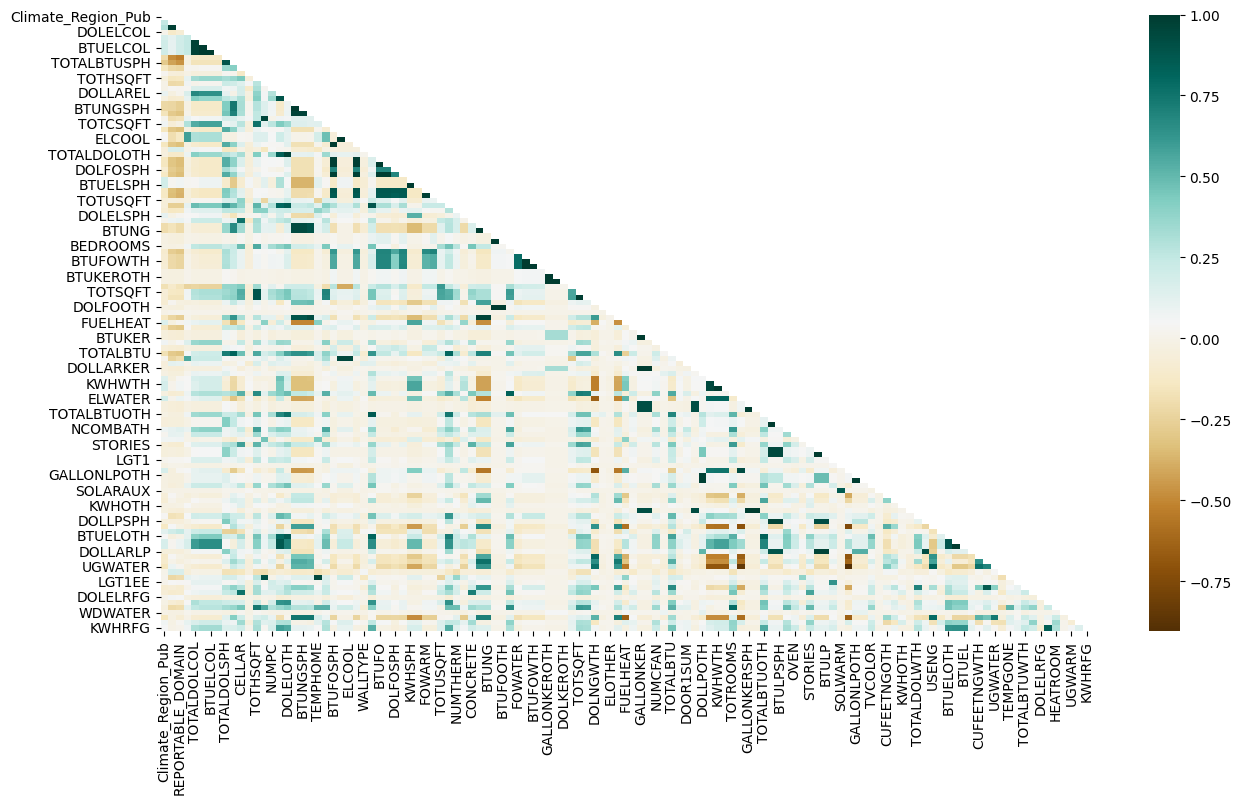


Рисунок 61 — Матрица корреляции

**Отбор признаков**

Отбор признаков для обучения был осуществлен с помощью библиотеки feature\_selector.

Ненужные признаки снижают скорость обучения модели, интерпретируемость и, главное, способность к обобщению. FeatureSelector – это специальный класс для отбора информативных признаков, написанный на языке Python, реализующий самые распространенные методы выборки. Его исходный код доступен на GitHub. Инструмент позволяет оценивать параметры по следующим характеристикам:

* высокий процент пропущенных значений;
* коллинеарность (сильная корреляция);
* нулевая важность в моделях, основанных на деревьях;
* низкая важность;
* единственное значение.

**Коллинеарные признаки**

Коллинеарные признаки в машинном обучении — это признаки, которые сильно коррелируют между собой, то есть они линейно зависимы.

Присутствие коллинеарных признаков может приводить к проблемам в алгоритмах машинного обучения. Например, в линейной регрессии или логистической регрессии коллинеарные признаки делают оценки коэффициентов неправильными или «шумными», что затрудняет интерпретацию влияния каждого признака на целевую переменную. Кроме того, коллинеарные признаки могут приводить к нестабильным результатам и переобучению модели.

Метод identify\_collinear находит пары коллинеарных признаков на основе коэффициента корреляции Пирсона. Для каждой пары, превышающей указанный порог (в пересчете на абсолютное значение), он идентифицирует одну из переменных, подлежащих удалению.

Для каждой пары признак, который будет удален, является последним с точки зрения упорядочения столбцов во фрейме данных.

Коллинеарные признаки представлены на рисунке 62.

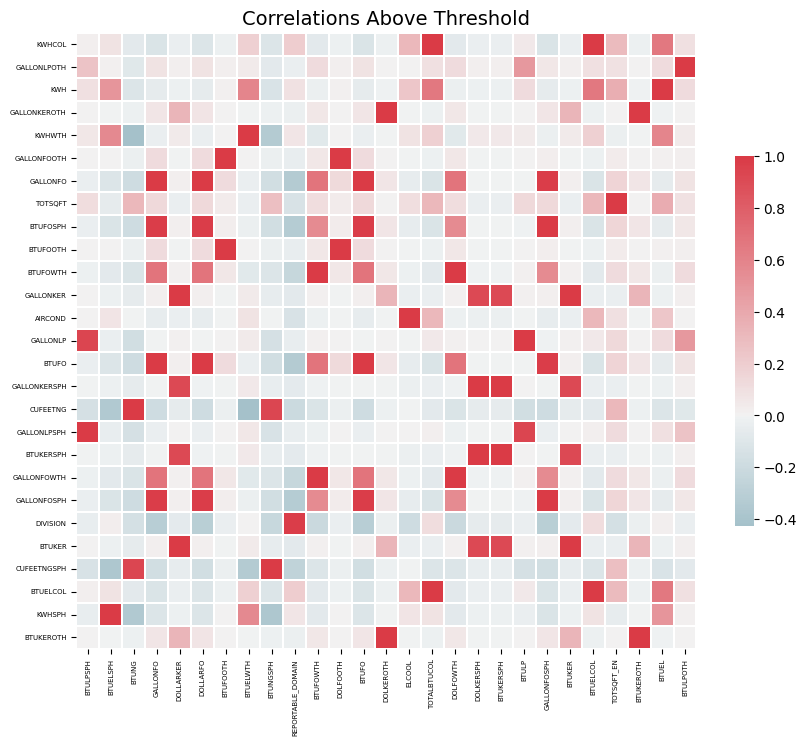


Рисунок 62 — Коллинеарные признаки

**Признаки нулевой важности**

Признаки нулевой важности в машинном обучении - это признаки, которые не имеют значимого влияния на целевую переменную или на результаты модели. Это означает, что удаление этих признаков из набора данных не должно значительно изменить качество модели или ее предсказательные способности.

Определение признаков нулевой важности может быть полезным для различных задач, таких как отбор признаков (feature selection), оптимизация модели, ускорение обучения модели и уменьшение размерности данных. Также, исключение ненужных признаков может снизить риск переобучения модели на «шумных» или нерепрезентативных данных.

На рисунке 63 представлена важность признаков.

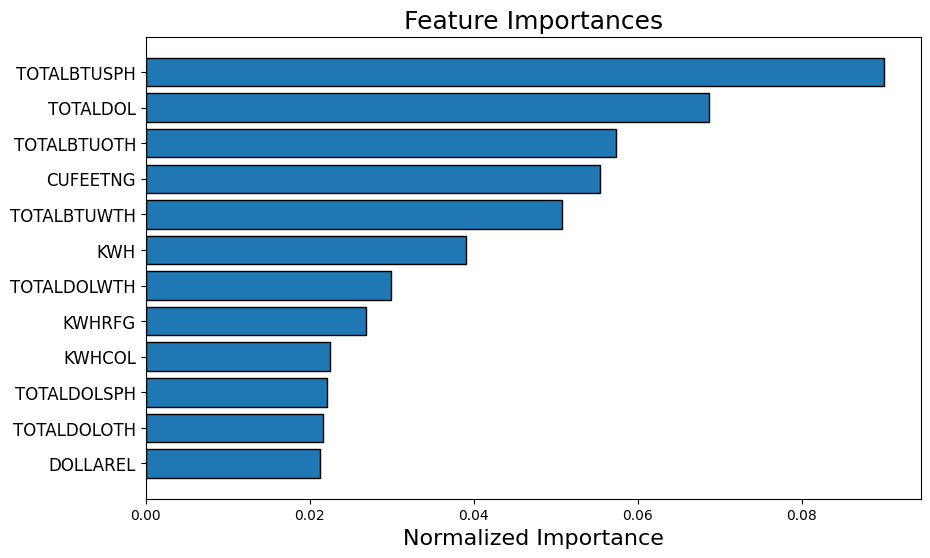


Рисунок 63 — Кумулятивная важность признаков

**Признаки с константным значением**

Признаки с константным значением в машинном обучении - это признаки, у которых значение одинаково для всех объектов в обучающей выборке.

Такие признаки несут мало или никакой информации, поскольку не изменяются в разных наблюдениях. Они могут вызывать проблемы в алгоритмах машинного обучения, особенно в линейных моделях. В таких моделях, например, признаки с константным значением могут привести к переобучению или невозможности корректно оценить веса признаков.

**Использованные признаки**

Таким образом, для построения прогноза были использованы следующие признаки:

* + 'Climate\_Region\_Pub',
  + 'DIVISION',
  + 'DOLELCOL',
  + 'TOTALDOLCOL',
  + 'KWHCOL',
  + 'TOTALDOLSPH',
  + 'TOTALBTUSPH',
  + 'CELLAR',
  + 'NWEIGHT',
  + 'TOTHSQFT',
  + 'NUMPC',
  + 'DOLLAREL',
  + 'DOLELOTH',
  + 'CUFEETNGSPH',
  + 'DOLNGSPH',
  + 'TEMPHOME',
  + 'TOTCSQFT',
  + 'BTUFOSPH',
  + 'WALLTYPE',
  + 'TOTALDOLOTH',
  + 'DOLFOSPH',
  + 'KWHSPH',
  + 'TOTUSQFT',
  + 'TOTALDOL',
  + 'NUMTHERM',
  + 'DOLELSPH',
  + 'CONCRETE',
  + 'CUFEETNG',
  + 'BEDROOMS',
  + 'FOWATER',
  + 'GALLONFOWTH',
  + 'GALLONKEROTH',
  + 'TOTUCSQFT',
  + 'TOTSQFT',
  + 'DOLNGWTH',
  + 'DOLLARNG',
  + 'FUELHEAT',
  + 'NOTMOIST',
  + 'GALLONKER',
  + 'NUMCFAN',
  + 'FOILAUX',
  + 'COOLTYPE',
  + 'DOOR1SUM',
  + 'DOLLPOTH',
  + 'DOLELWTH',
  + 'KWHWTH',
  + 'TOTROOMS',
  + 'GALLONKERSPH',
  + 'TOTALBTUOTH',
  + 'GALLONLPSPH',
  + 'NCOMBATH',
  + 'OVEN',
  + 'EQUIPM',
  + 'STORIES',
  + 'GALLONLP',
  + 'LGT1',
  + 'FUELH2O',
  + 'GALLONLPOTH',
  + 'TVCOLOR',
  + 'CUFEETNGOTH',
  + 'BTUNGOTH',
  + 'KWHOTH',
  + 'TOTALDOLWTH',
  + 'DOLLPSPH',
  + 'YEARMADE',
  + 'BTUELOTH',
  + 'KWH',
  + 'DOLLARLP',
  + 'CUFEETNGWTH',
  + 'BTUNGWTH',
  + 'SDESCENT',
  + 'TEMPGONE',
  + 'LGT1EE',
  + 'TOTALBTUWTH',
  + 'ROOFTYPE',
  + 'DOLELRFG',
  + 'TOTALDOLRFG',
  + 'HEATROOM',
  + 'DRYRFUEL',
  + 'KWHRFG'

4.3.3 Построенные нейросетевые модели

4.3.3.1 Модель №1

**Параметры архитектуры**

Архитектура данной нейронной сети состоит из двух слоев.

* Первый слой (Dense) имеет 32 нейрона и использует активационную функцию ReLU.
* Второй слой (Dense) состоит из одного нейрона и использует линейную активационную функцию.

Применение линейной функции активации имеет несколько преимуществ в задачах регрессии. Во-первых, линейная функция активации позволяет легко интерпретировать веса модели и понять, как каждый входной признак влияет на выходную переменную. Во-вторых, линейные модели обладают хорошей обобщающей способностью и могут быть легко обучены на больших наборах данных. В-третьих, линейная функция активации имеет простую аналитическую формулу для вычисления градиента, что упрощает реализацию алгоритма обучения.

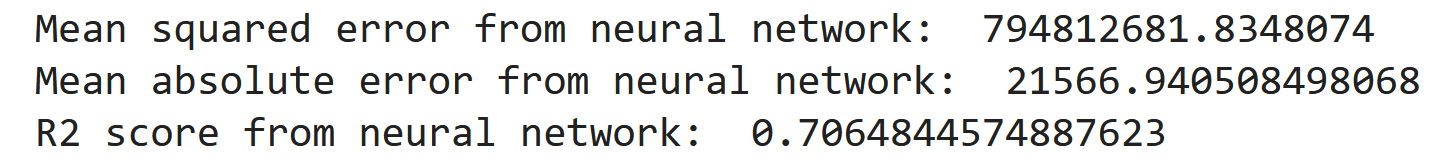
**Параметры обучения**

Параметры для обучения нейронной сети:

* Loss (функция потерь) - 'mean\_squared\_error' (среднеквадратичная ошибка) определяет способ измерения ошибки между предсказанными значениями и истинными метками.
* Optimizer (оптимизатор) - 'adam' определяет алгоритм оптимизации, который используется для обновления весов и улучшения производительности нейронной сети в процессе обучения.
* Metrics (метрики) - ['mae', 'mse'] задают метрики, которые будут отображаться в процессе обучения и оценки производительности модели. В данном случае использованы средняя абсолютная ошибка (MAE) и среднеквадратичная ошибка (MSE).

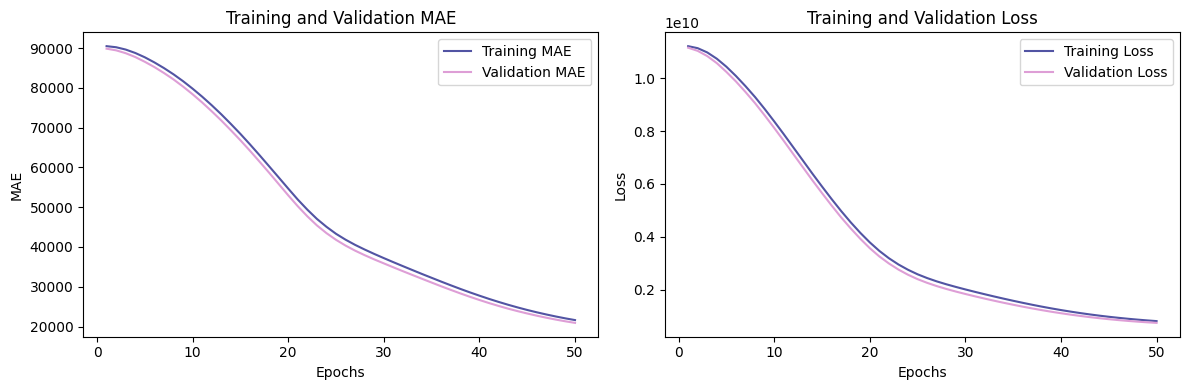
**Полученные метрики**

Метрики для модели представлены на рисунке 64.

Рисунок 64 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 65).

****Рисунок 65 — График визуализации обучения

Оценка предсказаний (рисунок 66).

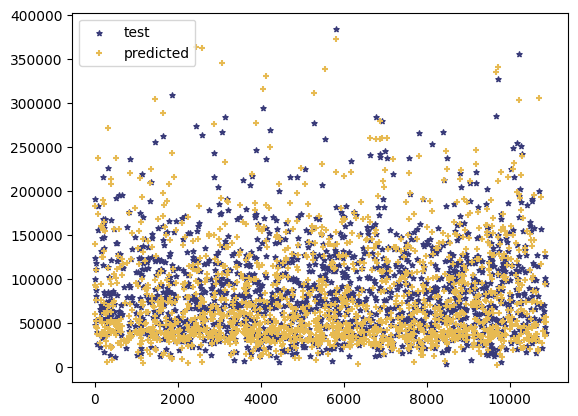
****

Рисунок 66 — Оценка предсказаний

4.3.3.2 Модель №2

**Параметры архитектуры**

Данная нейронная сеть имеет следующую архитектуру:

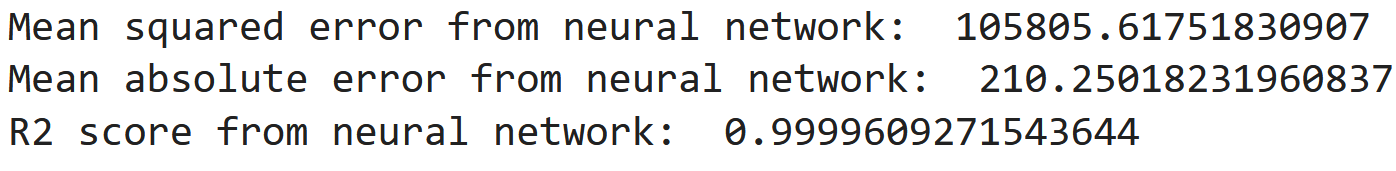
* Первый слой Dense: имеет 256 нейронов и использует функцию активации ReLU (Rectified Linear Unit).
* Второй слой Dense: имеет 128 нейронов и также использует функцию активации ReLU.
* Выходной слой: состоит из одного нейрона и использует линейную функцию активации.

**Параметры обучения**

Параметры обучения такие же, как и в предыдущей модели.

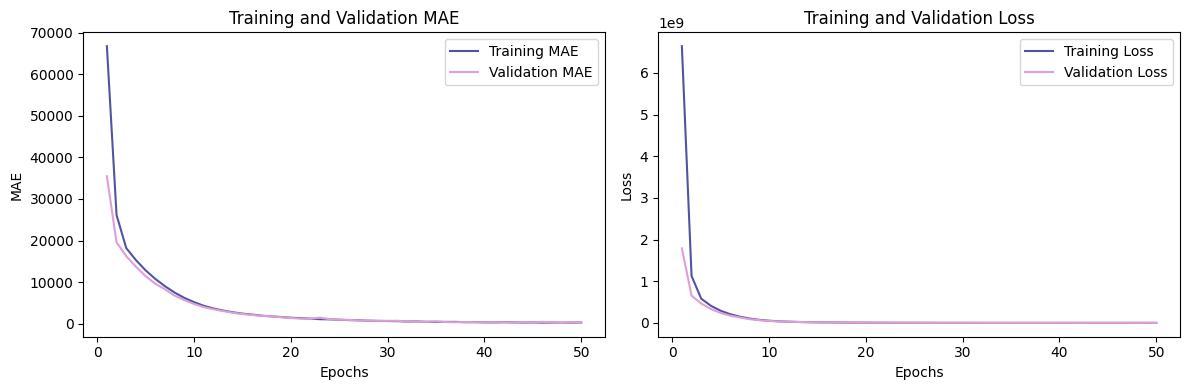
**Полученные метрики**

Метрики для модели представлены на рисунке 67.

Рисунок 67 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 68).

****Рисунок 68 — График визуализации обучения

Оценка предсказаний (рисунок 69).

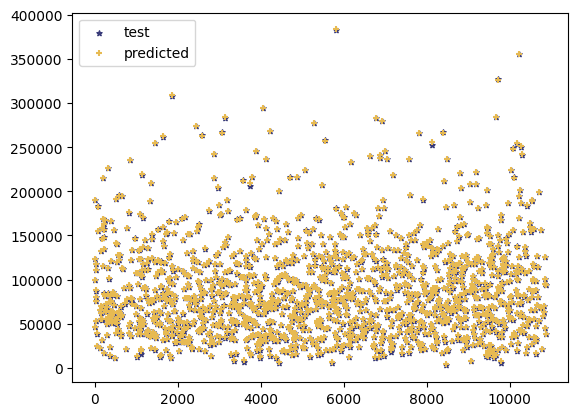
****

Рисунок 69 — Оценка предсказаний

4.3.3.3 Модель №3

**Параметры архитектуры**

Сохранена архитектура модели №2.

**Параметры обучения**

Оптимизатор заменен на RMSprop.

**Полученные метрики**

Метрики для модели представлены на рисунке 70.



Рисунок 70 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 71).

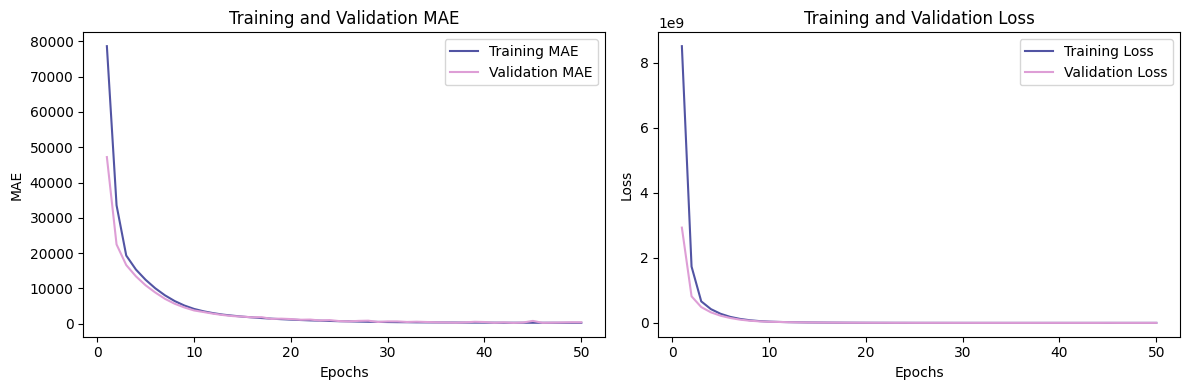
****

Рисунок 71 — График визуализации обучения

Оценка предсказаний (рисунок 72).

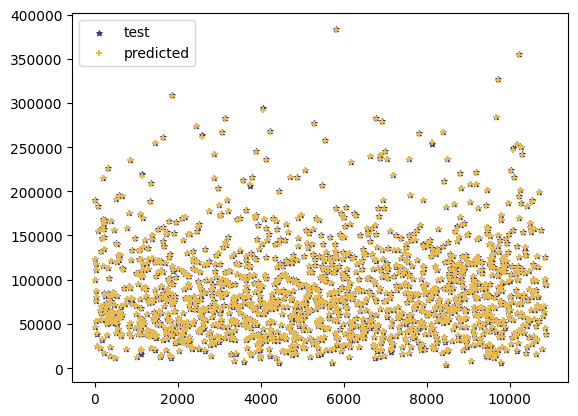
****

Рисунок 72 — Оценка предсказаний

4.3.3.4 Модель №4

**Параметры архитектуры**

Архитектура данной нейронной сети состоит из четырех слоев:

* Первый слой имеет 512 нейронов и активацию selu. Selu - это функция активации, которая обеспечивает нормализацию и стабилизацию градиента в слоях нейронной сети.
* Второй слой имеет 256 нейронов и также использует активацию selu.
* Третий слой состоит из 128 нейронов с активацией selu.
* Последний слой имеет один нейрон и активацию linear.

**Параметры обучения**

Используется оптимизатор Adam.

**Полученные метрики**

Метрики для модели представлены на рисунке 73.

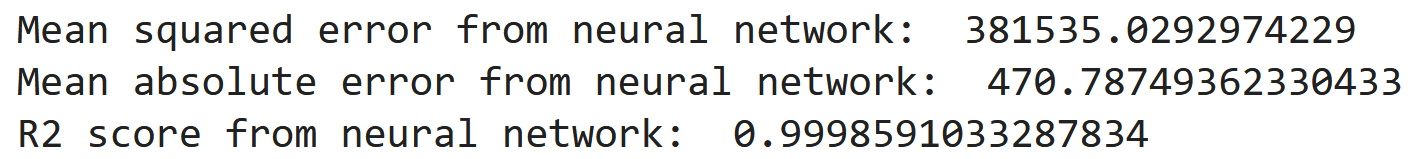


Рисунок 73 — Метрики

**Графики обучения**

График визуализации обучения (рисунок 74).

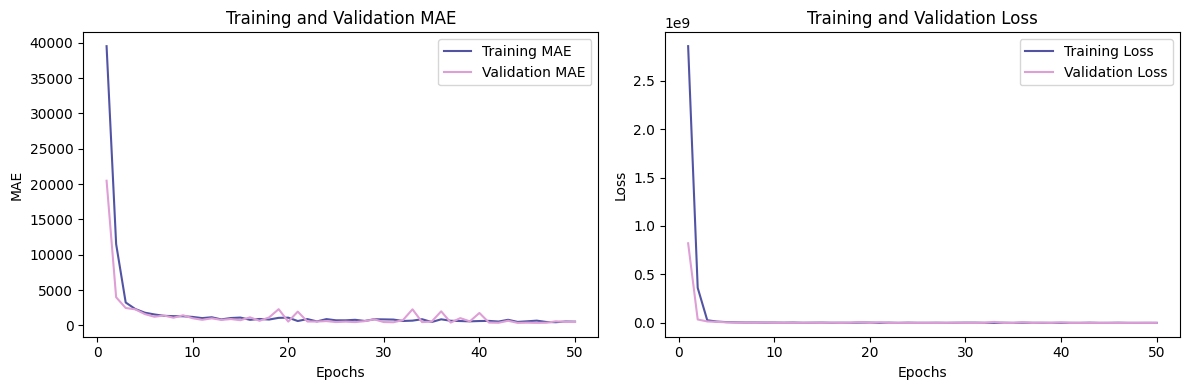
****

Рисунок 74 — График визуализации обучения

Оценка предсказаний (рисунок 75).

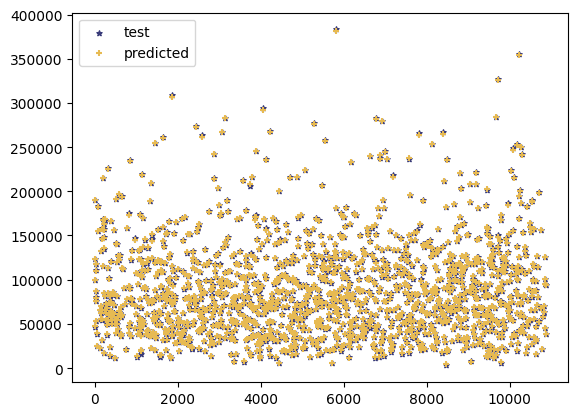
****

Рисунок 75 — Оценка предсказаний

4.3.4 Выводы по задаче регрессии

Каждая модель продемонстрировала очень высокую точность предсказаний. Немного лучше, чем остальные модели, результат продемонстрировала Модель №2, которая обладает довольно простой архитектурой.

# 5 Выводы

В результате выполнения задания был построен ряд нейросетевых архитектур, решающих задачи бинарной классификации, многоклассовой классификации и регрессии, и проведен их сравнительный анализ на основе заданных метрик. Большинство моделей продемонстрировали высокие результаты обучения.