ОТЧЕТ

Здание№1  
Вариант Чётный

«Полносвязные нейронные сети прямого распространения»

Студентка группы №932001: Ли Алина

**Цель работы** – получить навыки создания полносвязного бинарного и многоклассового классификаторов, а также нейросетевого регрессора с помощью библиотеки построения нейронных сетей.

**Задачи** – Построить три нейросетевые модели:

**a) 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').

**b) Многоклассовый классификатор**Оценка здоровья внутриутробного развития плода: <https://www.kaggle.com/datasets/andrewmvd/fetal-health-classification> Метка класса: fetal\_health.

**c) 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).

**Бинарный Классификатор**

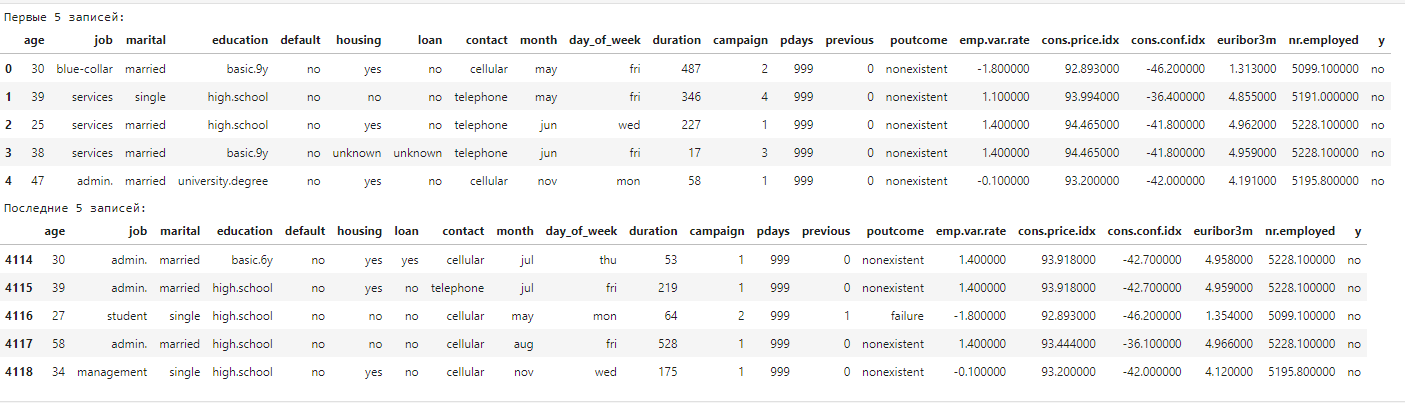
Целью данной задачи является построение бинарного классификатора для оценки вероятности того, что клиент откроет банковский депозит в результате маркетинговой акции.  
Класс "yes" будет означать, что клиент согласился открыть депозит, а класс "no" - что клиент отказался.

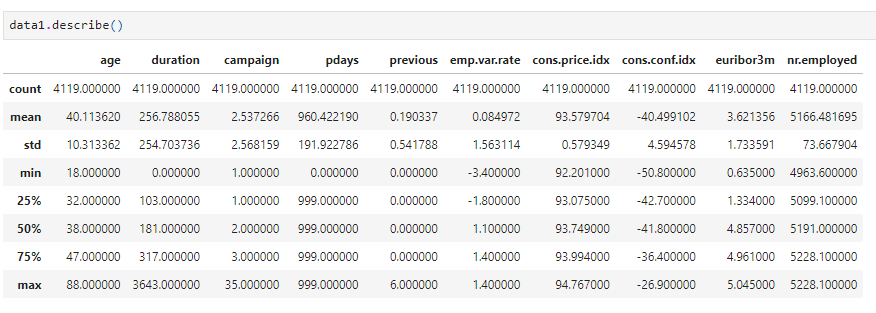
Исходные данные:

1. **age** - Возраст клиента (числовой).
2. **job** - Тип занятости клиента (категориальный).  
   тип работы (категориальная: 'admin.','blue-collar','entrepreneur','housemaid','management','retired','self-employed','services','student','technician','unemployed','unknown')
3. **marital** - Семейное положение клиента (категориальный).  
   семейное положение (категория: "разведен", "женат", "холост", "неизвестно"; примечание: "разведен" означает разведенный или овдовевший)
4. **education** - Уровень образования клиента (категориальный).  
   (категориальный: 'basic.4y','basic.6y','basic.9y','high.school','illiterate','professional.course','university.degree','unknown')
5. **default** - Наличие кредита в дефолте (бинарный: "yes", "no").  
   есть ли дефолт по кредиту?
6. **balance** - Средний годовой баланс клиента в евро (числовой).  
   среднегодовой баланс.
7. **housing** - Наличие ипотеки у клиента (бинарный: "yes", "no").  
   есть ли ипотека?
8. **loan** - Наличие личного займа у клиента (бинарный: "yes", "no").  
   есть личный заем?
9. **contact** - Тип коммуникации при контакте с клиентом (категориальный).  
   тип контактной связи (категория: "сотовая связь", "телефон")
10. **day\_of\_week** - День недели последнего контакта (категориальный).

последний контактный день недели

1. **month** - Месяц последнего контакта (категориальный).  
   месяц последнего контакта в году (категория: "январь", "февраль", "март", ..., "ноябрь", "декабрь")
2. **duration** - Продолжительность последнего контакта в секундах (числовой).
3. **campaign** - Количество контактов, проведенных в рамках текущей кампании и для данного клиента (числовой).
4. **pdays** - Количество дней, прошедших после последнего контакта клиента по предыдущей кампании (числовой, -1 означает, что клиент не был предварительно контактирован).
5. **previous** - Количество контактов, проведенных до текущей кампании и для данного клиента (числовой).
6. **poutcome** - Результат предыдущей маркетинговой кампании (категориальный).





Проанализируем некоторые из этих статистических показателей для нескольких признаков:  
**age** (Возраст клиентов):

Средний возраст клиентов составляет около 40 лет.

Минимальный возраст в наборе данных составляет 18 лет.

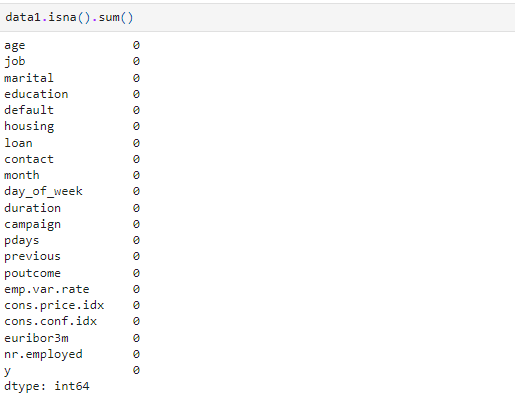
Максимальный возраст в наборе данных составляет 88 лет.

**duration** (Продолжительность последнего контакта):

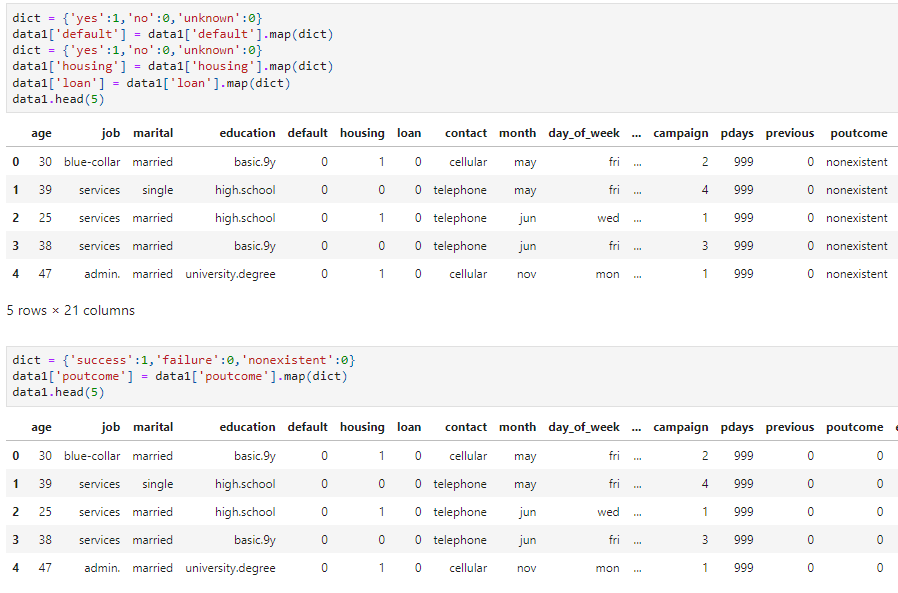
Средняя продолжительность последнего контакта составляет около 257 секунд.

Минимальная продолжительность контакта равна 0 секунд (возможно, это означает, что клиенты не были предварительно контактированы).

Максимальная продолжительность контакта составляет 3643 секунд.



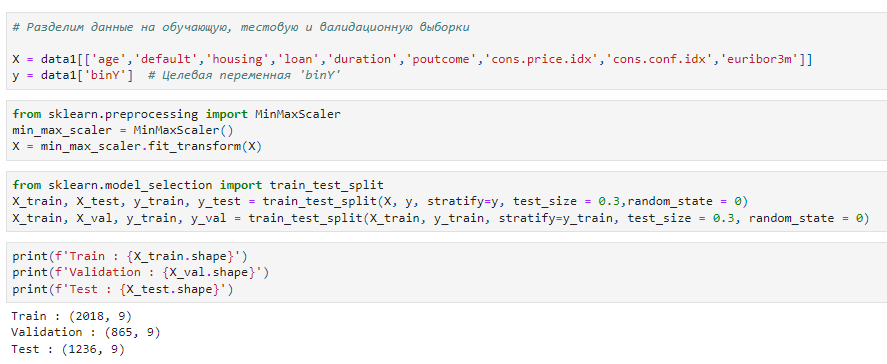
Результат выполнения этой команды равен нулю для всех столбцов, это означает, что в данных нет пропущенных значений и все столбцы полностью заполнены.



В коде происходит маппинг (перевод) категориальных значений в числовые значения для признаков 'default', 'housing' и 'loan'. Эти преобразования помогут работать с этими категориальными данными в алгоритмах машинного обучения, которые требуют числовых значений.  
Преобразуем также целевую переменную:



**Разделим данные на обучающую, тестовую и валидационную выборки**



**X** содержит выбранные признаки для обучения модели.   
Признаки: 'age', 'default', 'housing', 'loan', 'duration', 'poutcome', 'cons.price.idx', 'cons.conf.idx' и 'euribor3m'.

**y** представляет собой целевую переменную, которую нужно предсказать с использованием модели. В данном случае целевой переменной является 'binY'.  
**Нормализация признаков:** MinMaxScaler, отмасштабировали (нормализовали) значения признаков. Нормализация важна, так как она позволяет признакам находиться в одном и том же диапазоне значений, что помогает модели обучаться более эффективно.

**Разделение на обучающую, валидационную и тестовую выборки:**

X\_train и y\_train содержат обучающие данные и соответствующие им целевые значения. Эти данные будут использоваться для обучения модели.

X\_val и y\_val представляют валидационные данные, которые используются для настройки гиперпараметров модели и оценки ее производительности.

X\_test и y\_test представляют тестовые данные, которые используются для окончательной оценки производительности модели.  
Параметр stratify помогает сохранить баланс классов в разбиении выборки.

**Создание и обучение нейронной сети**

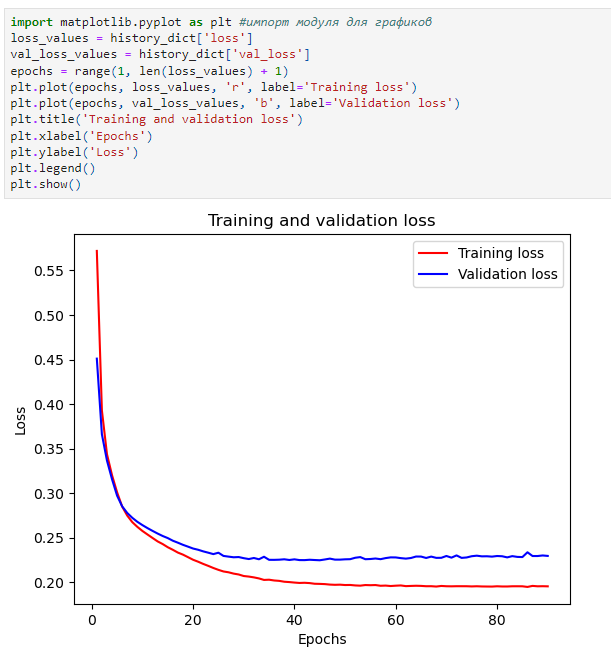
Создаем модель бинарной классификации с использованием библиотеки Keras.

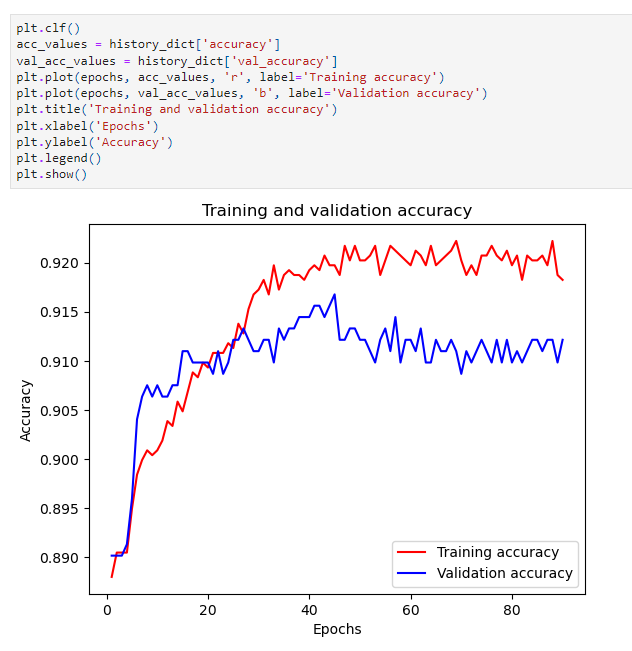


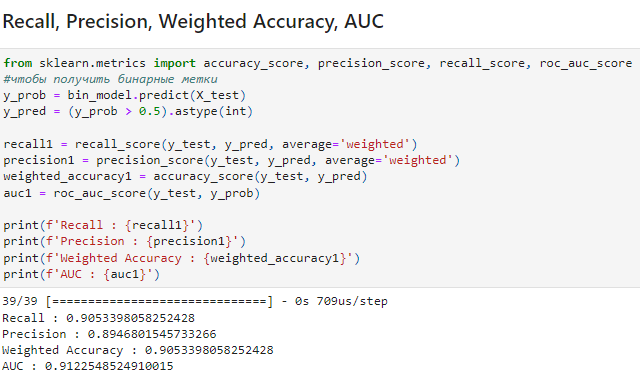
Первый слой с 4 нейронами, функцией активации ReLU и указанием входной формы input\_shape, которая определяется на основе числа признаков в X\_train.

Второй слой с 1 нейроном и функцией активации Sigmoid. Это слой вывода, который предсказывает бинарный результат.

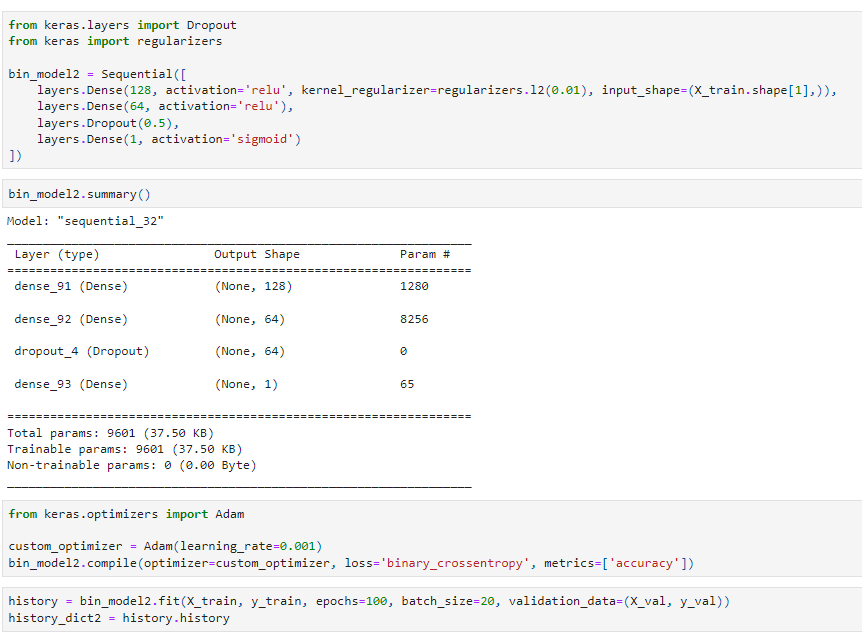
Компилируем модель, указывая оптимизатор adam, функцию потерь binary\_crossentropy, которая обычно используется для бинарной классификации и метрику для оценки производительности модели accuracy.  
Результаты обучения сохраняются в history\_dict.

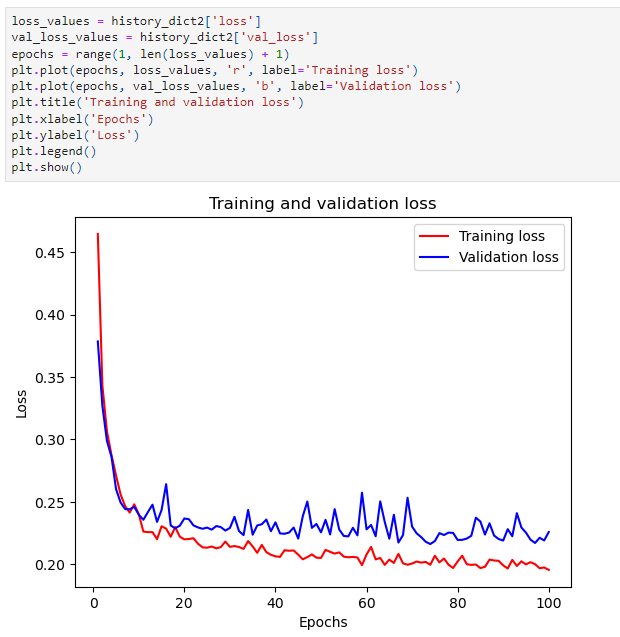


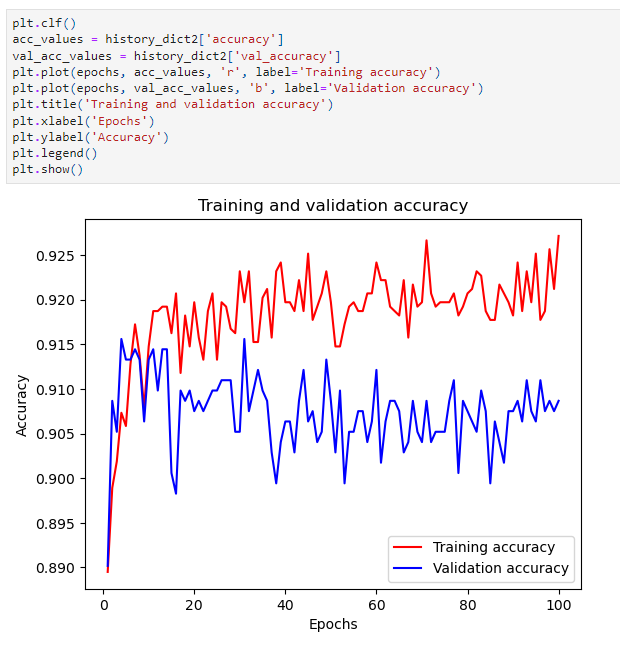


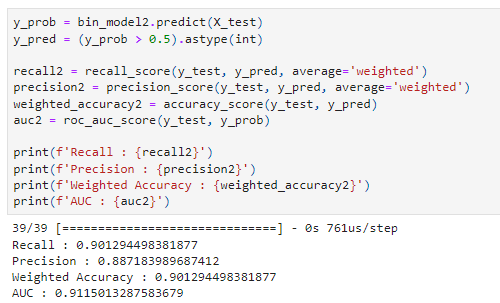


**Модель№2**

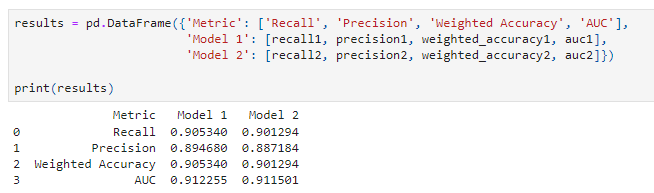
****

****

****

****

**Вывод:**

****

Анализ результатов оценки моделей:

Recall: Обе модели имеют высокие значения полноты, т.е они хорошо способны идентифицировать положительные случаи. (Model 1 немного превосходит Model 2)

Precision: Модель 1 имеет немного более высокую точность.

Weighted Accuracy: Обе модели показывают схожую взвешенную точность, что свидетельствует о хорошем балансе между полнотой и точностью.

AUC: Значения AUC для моделей близки друг к другу, что указывает на схожее качество моделей в разделении классов.

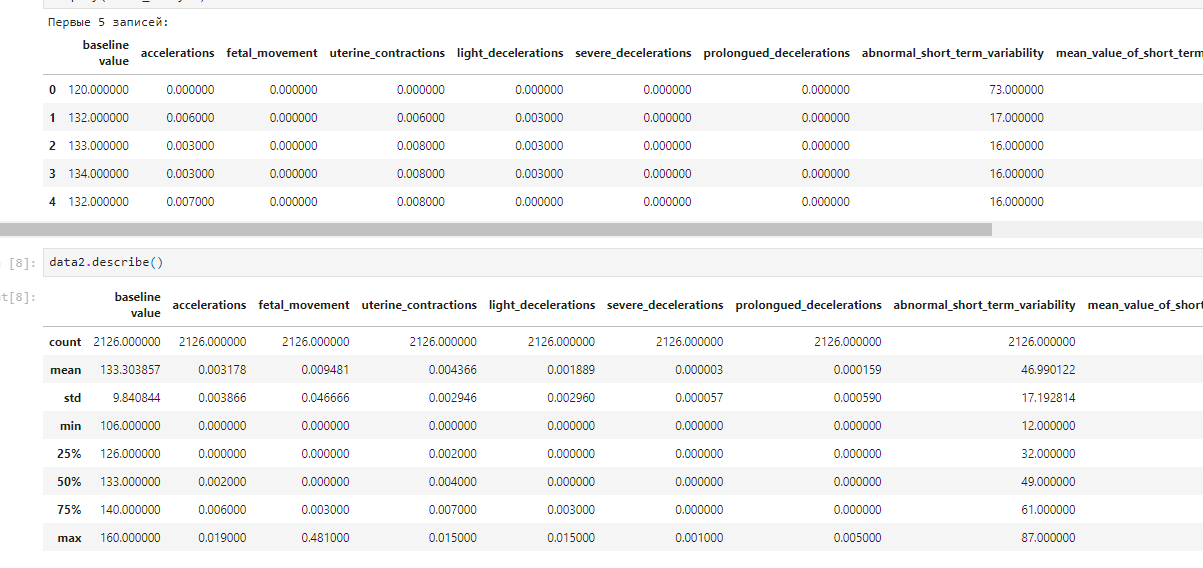
Обе модели представляют собой хорошие модели с высокими показателями полноты, точности и AUC.

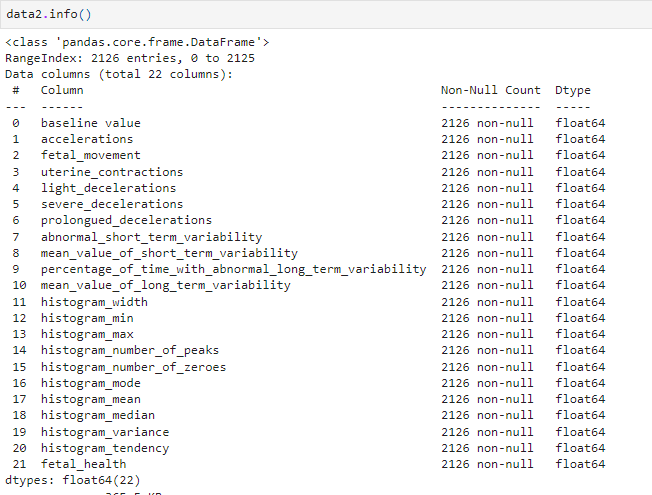
**Многоклассовый классификатор**

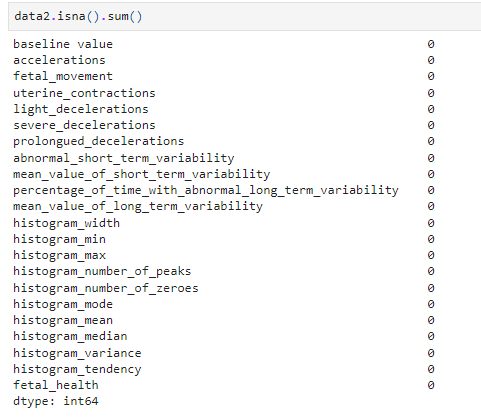
Цель: построение многоклассового классификатора, который будет способен предсказать состояние здоровья плода на основе предоставленных признаков (1 – Нормальное, 2 – Подозрение, 3 – Патологическое.)

Исходные данные:

1. **baseline value** базовая частота сердечных сокращений плода (FHR)
2. **accelerations** количество ускорений в секунду
3. **fetal\_movement** количество шевелений плода в секунду
4. **uterine\_contractions** количество сокращений матки в секунду
5. **light\_decelerations** количество легких замедлений в секунду
6. **severe\_decelerations** количество тяжелых замедлений в секунду
7. **prolongued\_decelerations** количество длительных замедлений в секунду
8. **abnormal\_short\_term\_variability** процент времени с аномальной краткосрочной изменчивостью
9. **mean\_value\_of\_short\_term\_variability** среднее значение краткосрочной изменчивости
10. **percentage\_of\_time\_with\_abnormal\_long\_term\_variability** процент времени с аномальной долгосрочной изменчивостью
11. **mean\_value\_of\_long\_term\_variability** среднее значение долгосрочной изменчивости
12. **histogram\_width** ширина гистограммы, составленной с использованием всех значений из записи.
13. **histogram\_min** минимальное значение гистограммы
14. **histogram\_max** максимальное значение гистограммы
15. **histogram\_number\_of\_peaks** количество пиков на гистограмме экзамена
16. **histogram\_number\_of\_zeroes** количество нулей в гистограмме экзамена
17. **histogram\_mode** мода гистограммы
18. **histogram\_mean** среднее значение гистограммы
19. **histogram\_median** медиан гистограммы
20. **histogram\_variance** отклонение гистограммы
21. **histogram\_tendency** тренд гистограммы
22. **fetal\_health** здоровье плода: 1 – Нормальное 2 – Подозрение 3 – Патологическое





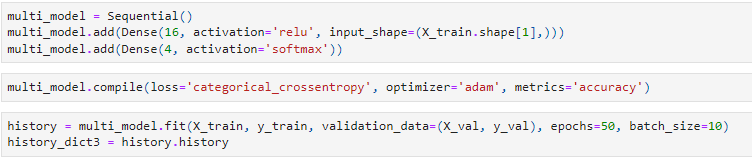


**Разделим данные на обучающую, тестовую и валидационную выборки**



Изначально формируем набор признаков (X), исключая целевую переменную "fetal\_health" из данных data2. Создаем целевую переменную (y), которую будем предсказывать, и она содержит состояния здоровья плода.  
MinMaxScaler - для масштабирования числовых признаков. Этот шаг помогает привести числовые значения признаков к интервалу [0, 1].  
Данные разделились на обучающую, валидационную и тестовую выборки. Далее целевую переменную "fetal\_health" преобразовали в формат one-hot encoding, что является необходимым для многоклассовой классификации.

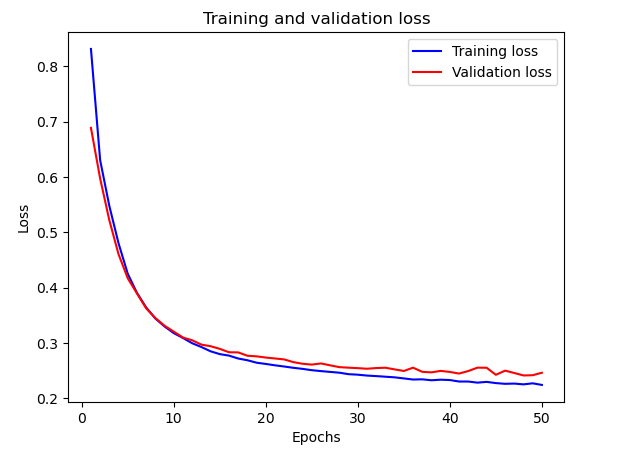
**Создание и обучение**

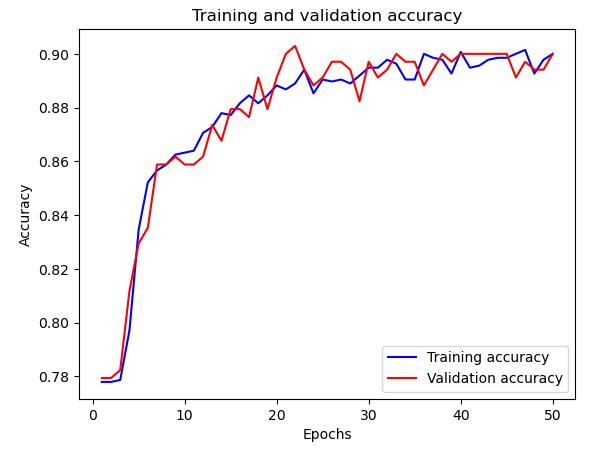
****

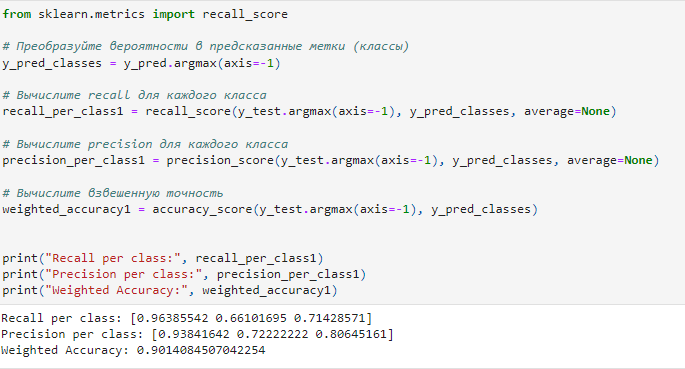
Добавляется полносвязный слой с 16 нейронами и функцией активации ReLU. Также указывается input\_shape, чтобы определить размер входных данных, равный количеству признаков в обучающем наборе.

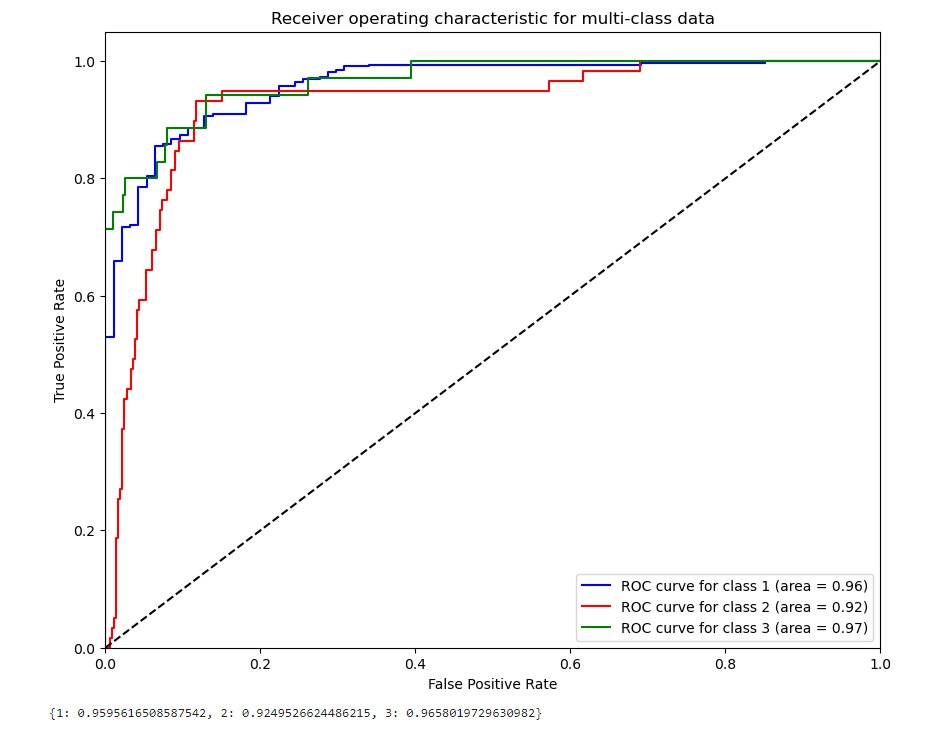
Добавляется второй полносвязный слой с 4 нейронами и функцией активации softmax. Это последний слой, который будет использоваться для многоклассовой классификации. Функция softmax обеспечивает вероятностные выводы для каждого класса.

Модель обучается в течение 50 эпох, и размер пакета (batch size) установлен на 10.

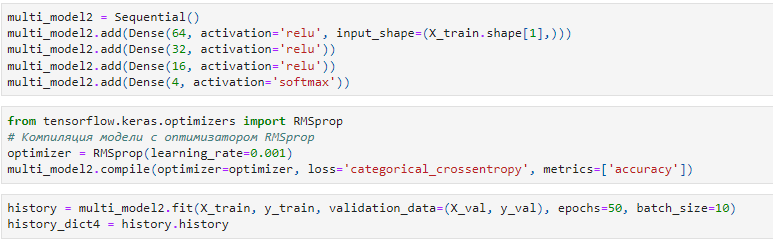








**Модель№2**



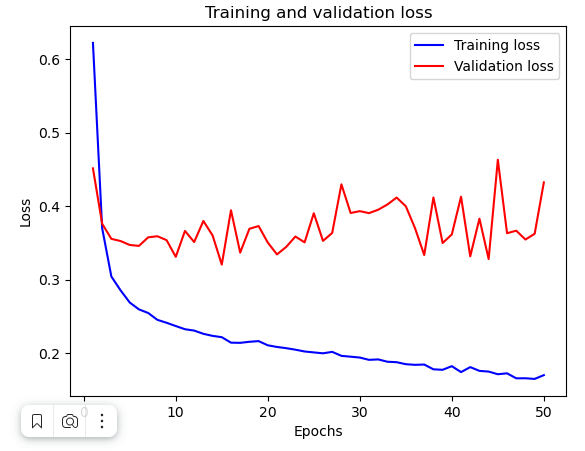
Добавляется первый полносвязный слой с 64 нейронами и функцией активации ReLU. Это служит входным слоем для модели.  
Затем добавляется второй полносвязный слой с 32 нейронами и функцией активации ReLU.

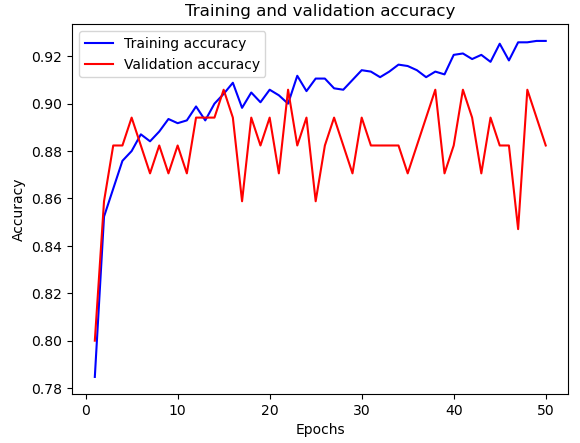
Третий полносвязный слой с 16 нейронами и функцией активации ReLU.

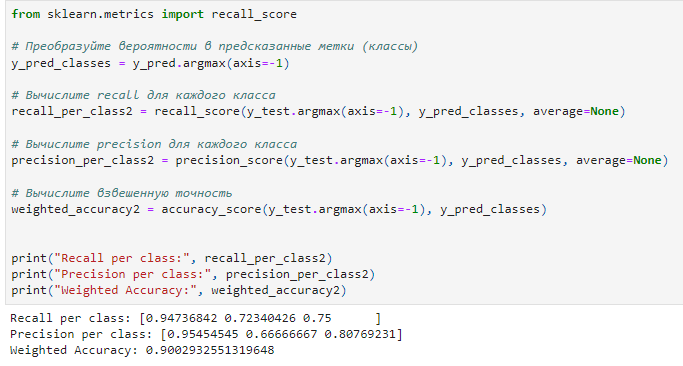
Заключительный слой с 4 нейронами и функцией активации softmax. Этот слой генерирует вероятностные выводы для классов.

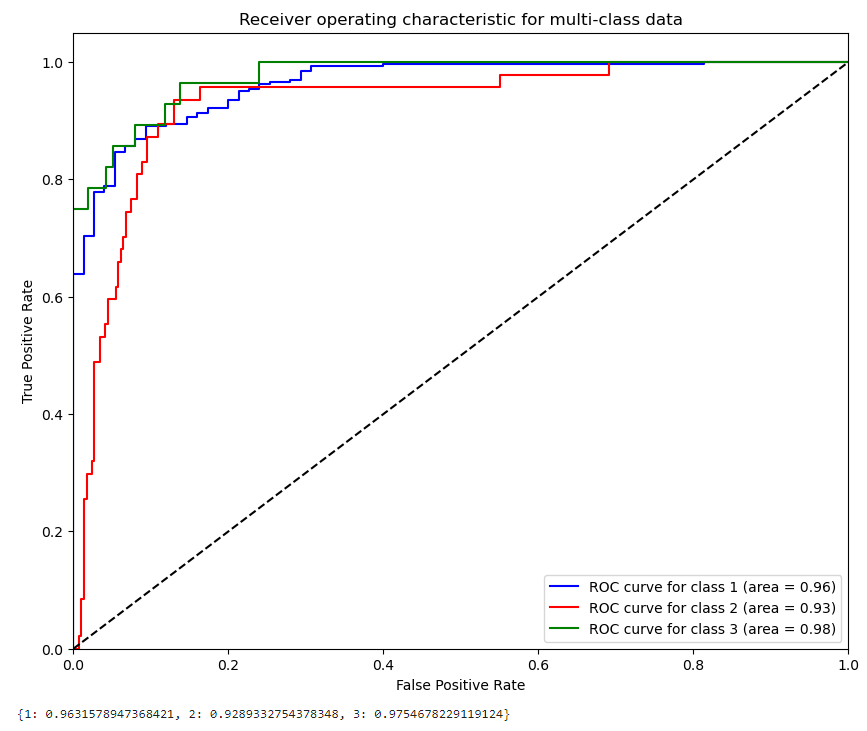
Далее создается экземпляр оптимизатора RMSprop с установленной скоростью обучения (learning rate) равной 0.001.

Далее происходит компиляция второй модели с использованием оптимизатора RMSprop. Функция потерь установлена как categorical\_crossentropy.









**Вывод:**

Обе модели имеют высокую чувствительность к классу 1 и довольно высокую точность для класса 1.

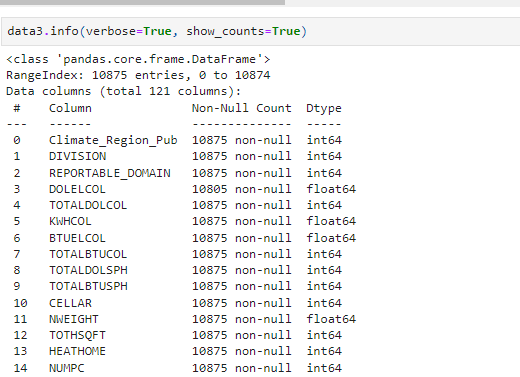
Значения взвешенной точности и AUC ROC примерно одинаковы для обеих моделей.

В целом, обе модели показывают хорошие результаты.

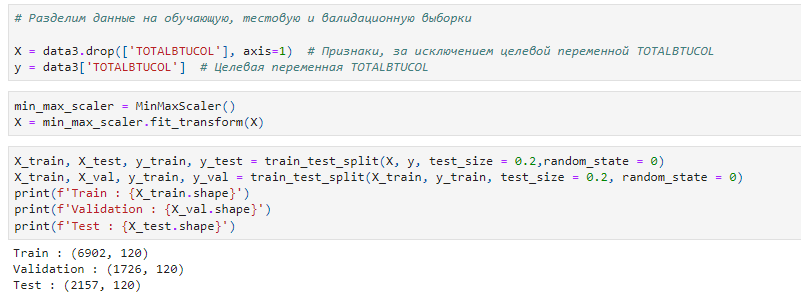
**Регрессор**

Цель - предсказать один из атрибутов, связанных с потреблением

энергии (TOTALBTUCOL).



**Разделим данные на обучающую, тестовую и валидационную выборки**

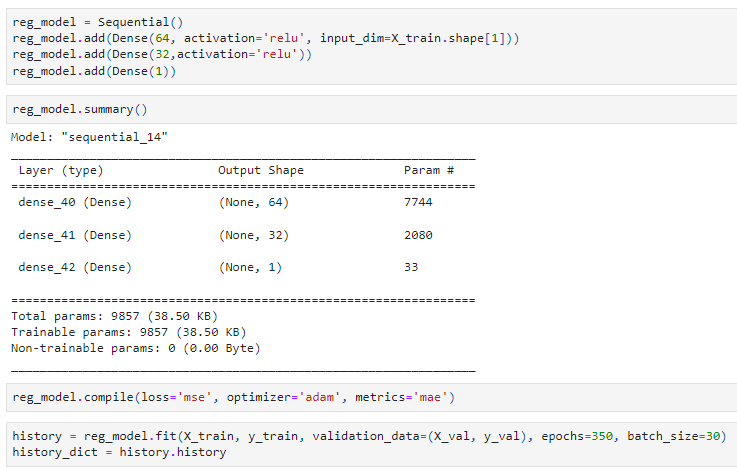


X представляет собой матрицу признаков, исключив из данных столбец с целевой переменной TOTALBTUCOL.

Переменная y содержит целевую переменную TOTALBTUCOL.

С помощью функции train\_test\_split данные были разделены на обучающий, валидационный и тестовый наборы.

**Создание и обучение**



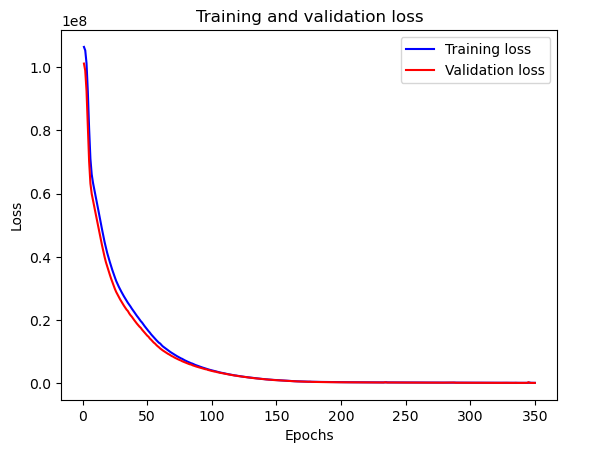
Первый слой с 64 нейронами и активацией ReLU.

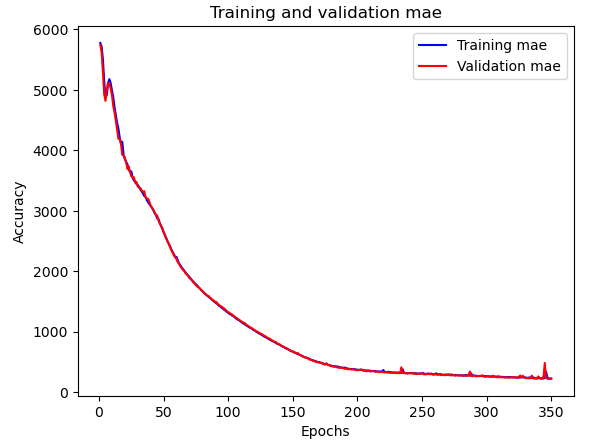
input\_dim устанавливается равным количеству признаков в обучающем наборе данных, так как это первый слой.

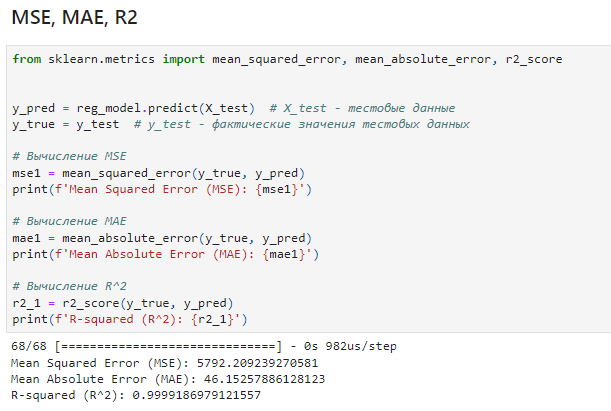
Второй слой с 32 нейронами и активацией ReLU.

Третий слой с одним нейроном, который будет использоваться для регрессии, поскольку вы пытаетесь предсказать числовое значение.  
Компиляция с использованием функции потерь MSE (Mean Squared Error) и оптимизатора adam. Также метрика MAE (Mean Absolute Error) для отслеживания качества обучения модели.

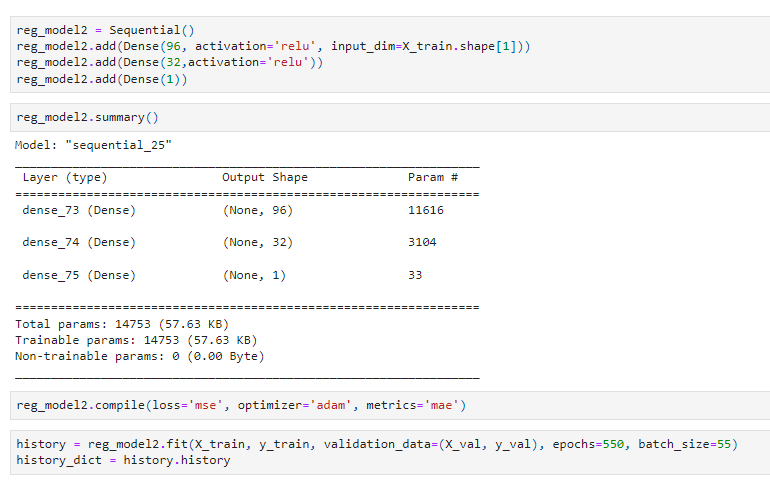
Обучение происходит на протяжении 350 эпох с размером пакета (batch\_size) равным 30.







**Модель№2**

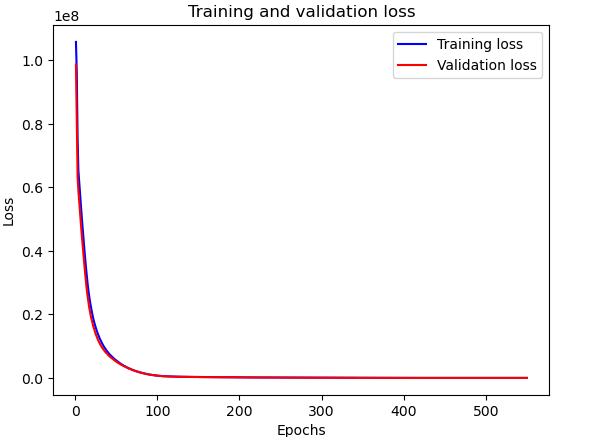


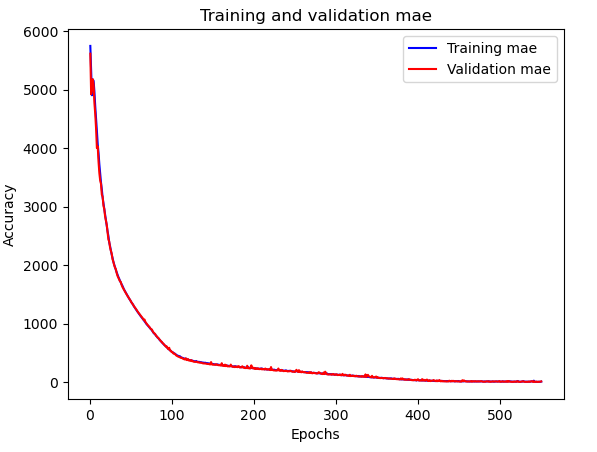
Первый слой с 96 нейронами и активацией ReLU.

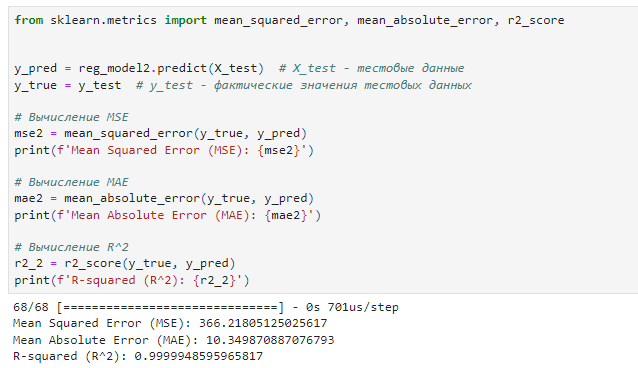
Второй слой с 32 нейронами и активацией ReLU.

Третий слой с одним нейроном для регрессии.

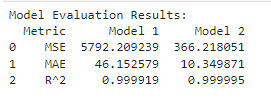
Обучение происходит на протяжении 550 эпох с размером пакета (batch\_size) равным 55.

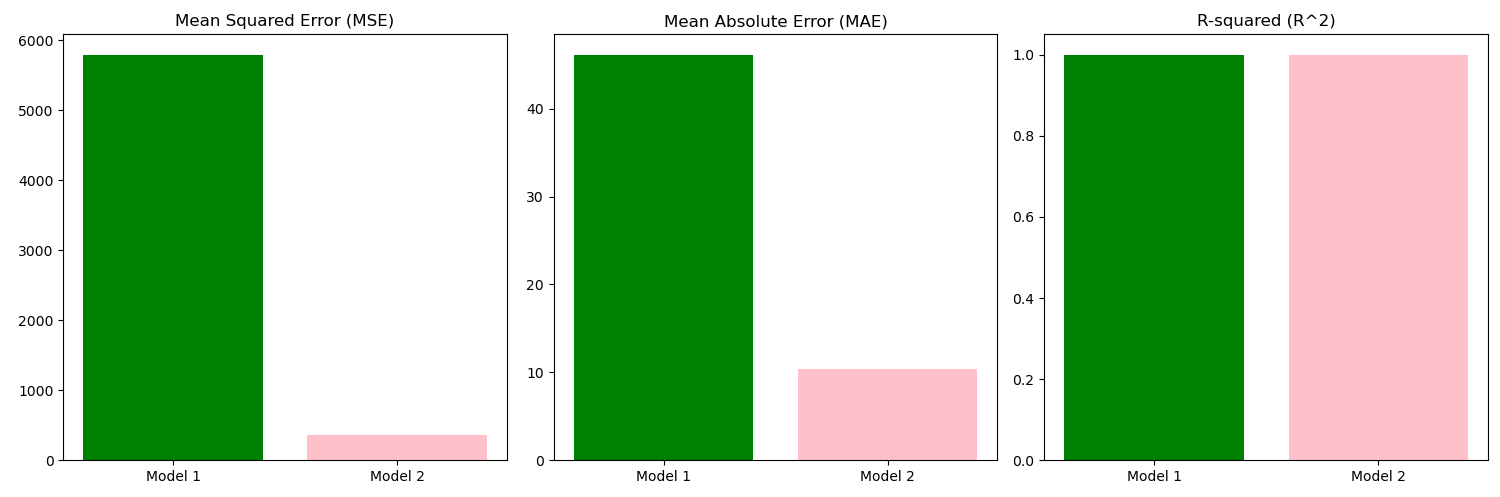






**Вывод:**

****

****

Анализ результатов оценки моделей:

MSE: Model2 имеет гораздо более низкое значение MSE, это говорит о том, что Model2 имеет лучшую точность прогнозирования, чем Model1. Чем меньше значение MSE, тем ближе модель к идеальной.

MAE: Model2 лучше справляется с прогнозированием, так как MAE меньше.

R^2: Обе модели имеют высокие значения R^2 (близкие к 1). Однако Model 2 имеет немного более высокое значение R^2, что указывает на более точное соответствие данным.

В общем, Model 2 превосходит Model 1 по всем метрикам оценки, следовательно, является более точной моделью для данной задачи прогнозирования.