

Оглавление

[**1.ВВЕДЕНИЕ** 2](#_Toc168760898)

[**2.** **Анализ предметной области** 4](#_Toc168760899)

**2.1 Подбор и установка среды разработки для работы с языком программирования Python……………………………………………………………………………………………………...**4

[**2.2** **Анализ теоретических аспектов создания нейросетей** 4](#_Toc168760900)

[**3.** **Обучение нейронной сети** 10](#_Toc168760901)

[**3.1** **Выбор способа оценки адекватности результатов обучения** 10](#_Toc168760902)

[**3.2** **Анализ адекватности обучения** 12](#_Toc168760907)

[**3.3** **Вывод по разделу** 13](#_Toc168760911)

[**ЗАКЛЮЧЕНИЕ** 14](#_Toc168760912)

**СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ**……………..……………………………………………………………...……………18

1. **ВВЕДЕНИЕ**

В Московском университете им. С.Ю. Витте на кафедре информационных систем проходила практика по производственной деятельности. Это обязательная часть обучения студентов всех специальностей, включая IT, которая помогает им получить практические навыки для будущей профессиональной деятельности. Во время практики студенты закрепляют и углубляют знания, приобретают навыки и опыт практической работы в своей специальности и квалификации.

**Цель практики**

Получение опыта работы с информационными технологиями в университете для решения задач и улучшения навыков выбора и использования подходящих технологий.

**Задачи практики**

- Закрепление теоретических знаний.

- Приобретение опыта создания нейронных сетей с использованием Python.

- Применение стандартов при решении практических задач.

- Формулирование требований к информационной системе после анализа предметной области.

- Создание математической модели для нейронной сети.

- Сбор и стандартизация данных для обучения нейронной сети.

- Обучение нейронной сети для достижения поставленных целей.

- Развитие навыков самостоятельной работы.

- Подготовка отчётов и презентаций на основе результатов профессиональной деятельности.

В качестве источников информации использовались организационно-правовые документы, распорядительные и информационно-справочные материалы, которые регламентируют работу специалиста в данной области.

1. **АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ**

Предметной областью данного проекта является использование нейронных сетей в предсказание стоимости автомобиля исходя из его характеристик.

**2.1 Подбор и установка среды разработки для работы с языком программирования Python**

В качестве инструмента для разработки проекта был выбран редактор кода VisualStudioCode. Он основан на быстром редакторе исходного кода и отлично подходит для ежедневного использования. Благодаря тому, что Visual Studio Code поддерживает множество языков программирования, он помогает повысить производительность пользователей. Это достигается за счёт таких функций, как подсветка синтаксиса, автоматический подбор скобок, выравнивание отступов, выделение блоков кода и использование фрагментов.

Конец формы

Простые сочетания клавиш, легкая настройка и поддержка от сообщества делают навигацию и работу с кодом простыми и интуитивными.

* 1. **Анализ теоретических аспектов создания нейросетей**

Нейронная сеть, также известная как искусственная нейронная сеть - это модель, созданная по образу и подобию биологических нейронных сетей нашего организма. Она была разработана для изучения процессов, происходящих в мозге, и их моделирования. После создания алгоритмов обучения нейронные сети стали использоваться в различных областях, таких как прогнозирование, распознавание образов и управление. Эта система состоит из соединенных между собой и взаимодействующих искусственных нейронов. Хотя отдельные процессоры в такой сети просты, их совместное взаимодействие позволяет выполнять сложные задачи. Нейронная сеть представляет собой частный случай методов машинного обучения, который широко используется в современных технологиях.

* 1. **Выбор набора данных**

Для обучения нейронной сети был выбран набор данных , содержащий в себе 17 признаков , но так стоит задача предсказать именно цену автомобиля , то разумно будет малорелевантные признаки убрать из выборки. Например, сбор и цвет очень слабо влияют на цену автомобиля , в отличие от пробега , года выпуска, коробки передач и тд. (см. рис. 1)

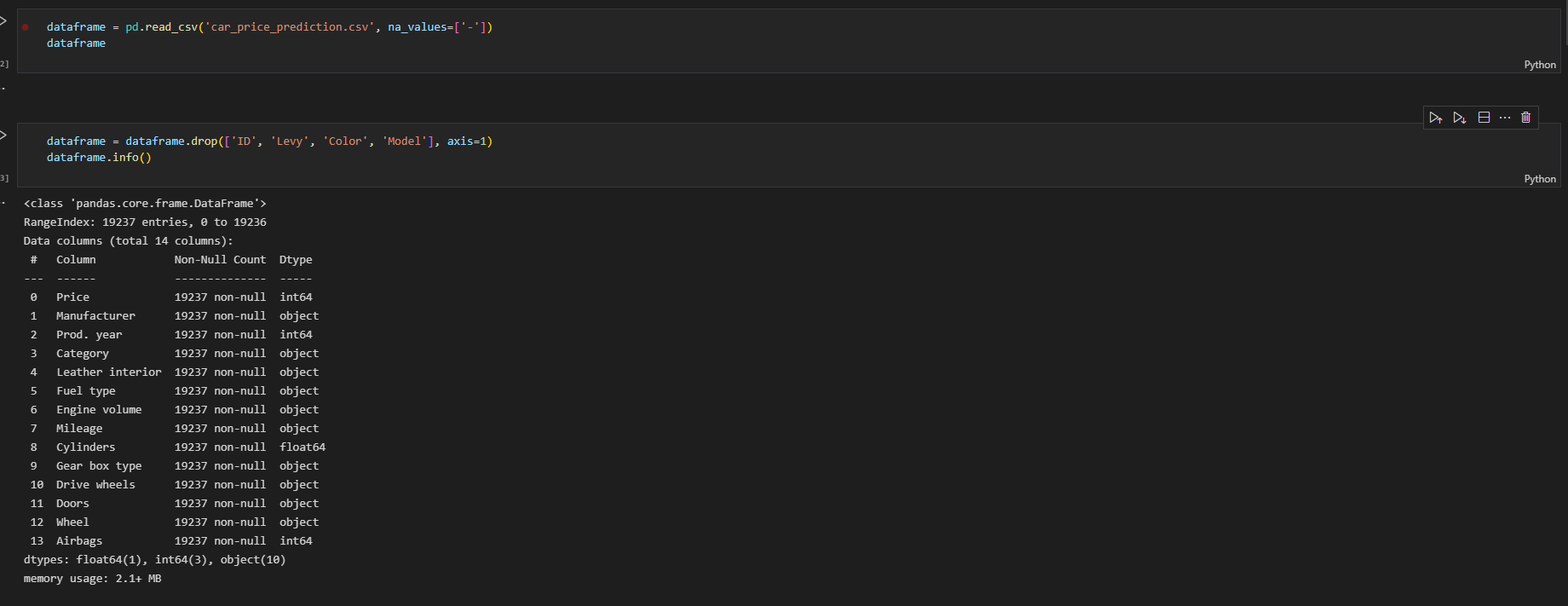


Рисунок 1 – Выбранные данные

Рисунок 1 – Выбранные данные

Так как минимальная цена автомобиля равна 1 доллару, исходя из логики ограничим нижнюю границу до 1000 долларов, а также мощность двигателя не должна равняться нулю (см. рис. 2)

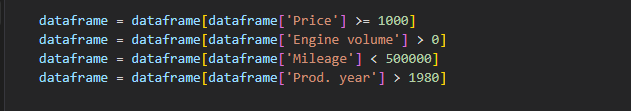


Рисунок 2 – Условия для данных

Выведем график «Ящик с усами» для определения выбросов относительно основного распределения (см. рис. 3)



Рисунок 3 – Ящик с усами

График показывает аномально большие выбросы, поэтому удалим их по следующей формуле (см. рис. 4)

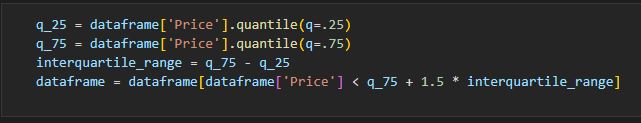
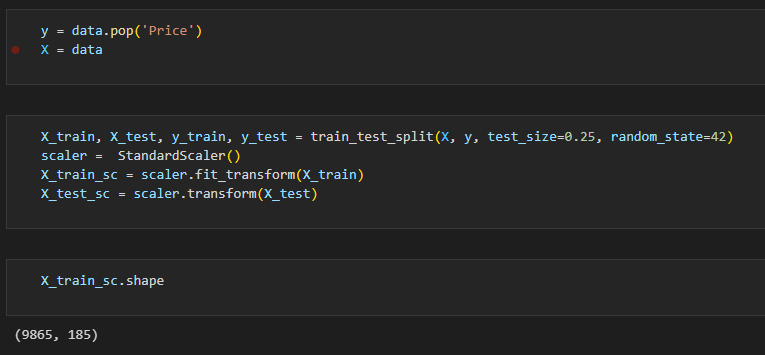


Рисунок 4 – формула удаления выбросов

* 1. **Выводы по разделу**

|  |  |  |
| --- | --- | --- |
| Выводы | Сформированные компетенции | |
| Код компетенции | Содержание компетенции |
| Проведен выбор и развертывание среды разработки для языка программирования Python. | ПК-9 | Навыки разработки методов сбора, анализа и обработки информации. |
| Проведен анализ теоретических аспектов создания нейросетей |
| Разработана математическая модель нейронной сети | ПК-10 | Способность применять математическое моделирование для анализа данных, используя инструменты машинного обучения и интеллектуального анализа. |

1. **ОБУЧЕНИЕ НЕЙРОННОЙ СЕТИ**

Мы применяем функцию train\_test\_split из библиотеки sklearn для того, чтобы разделить данные на обучающую и тестовую выборки.

Конец формы

Размер тестовой выборки составляет 25%, а для случайного перемешивания данных мы указываем значение 42. В итоге получаем две выборки – X\_train и X\_test. (см. рис. 5)

Рисунок 5 – распределение данных на обучающую и тренировочную части

Чтобы избежать переобучение модели нужно указать количество слоев и нейронов (см. рис. 6)

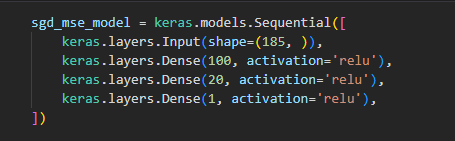
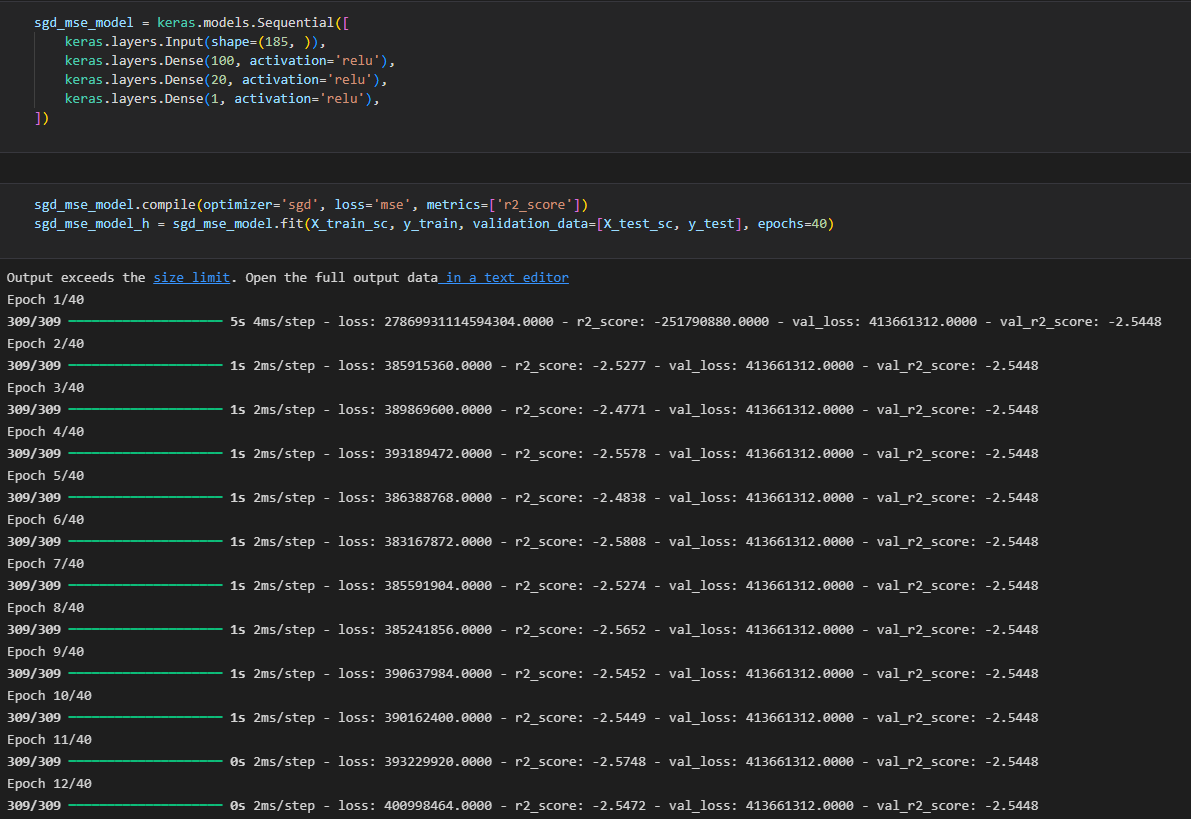


Рисунок 6 – Количество нейронов и слоев

Создадим первую модель с оптимизатором sgd и функцией потерь mse, она будет иметь 121 нейрон для обучения (см. рис. 7)

Рисунок 7 – первая модель

Скорость обучения первой модели занимает примерно 2 секунды и потери составляют минимально 383906048

Вторая модель которую мы обучим будет иметь оптимизатор adam, функцию потерь mae и будет иметь также 121 нейрон (см. рис. 8)

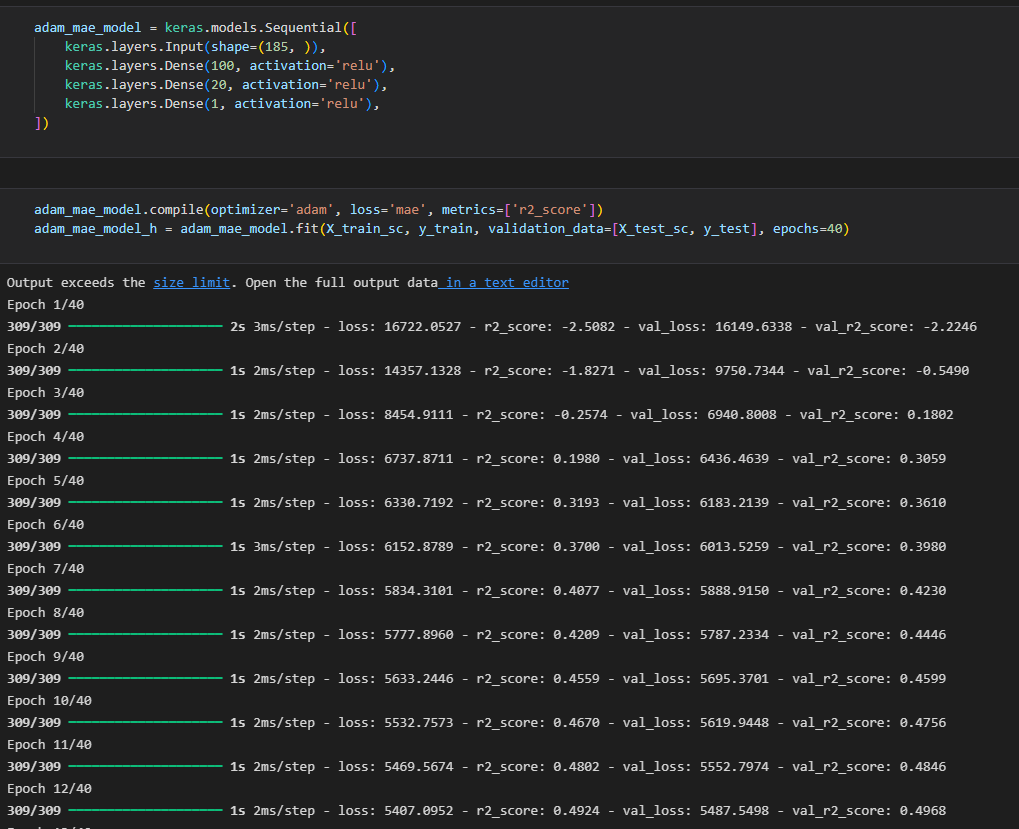


Рисунок 8 – Вторая модель

Скорость обучения второй модели занимает также как и первая примерно 2 секунды, но потери составляют уже 4466.8721

Третья модель будет иметь оптимизатор adam и функцию потерь mse , но в отличие от первых двух будет иметь 751 нейрон для обучения (см. рис. 9)

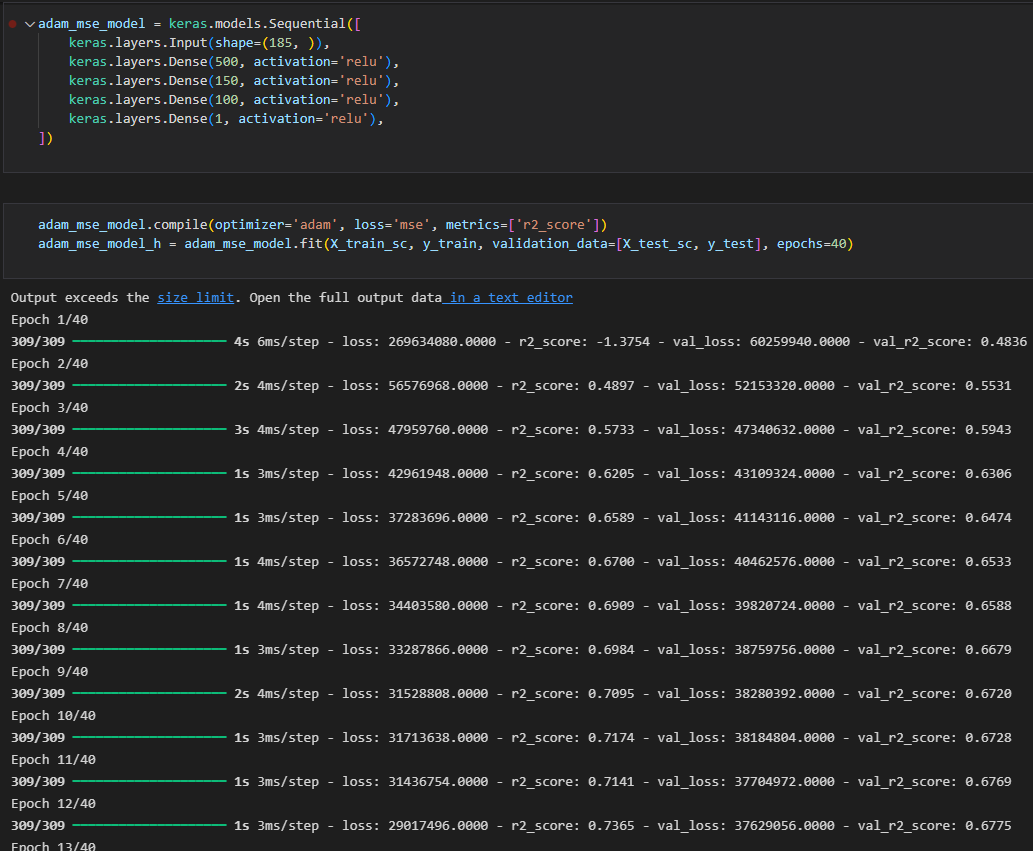


Рисунок 9 – Третья модель

Скорость обучения третьей модели выше чем у предыдущих моделей и составляет примерно 3-4 секунды, потери составляют минимально 19680882

Четвертая модель будет самое большое количество нейронов, а именно 1001 нейрон и с оптимизатором rmsprop и функцией потерь mse (см. рис. 10)

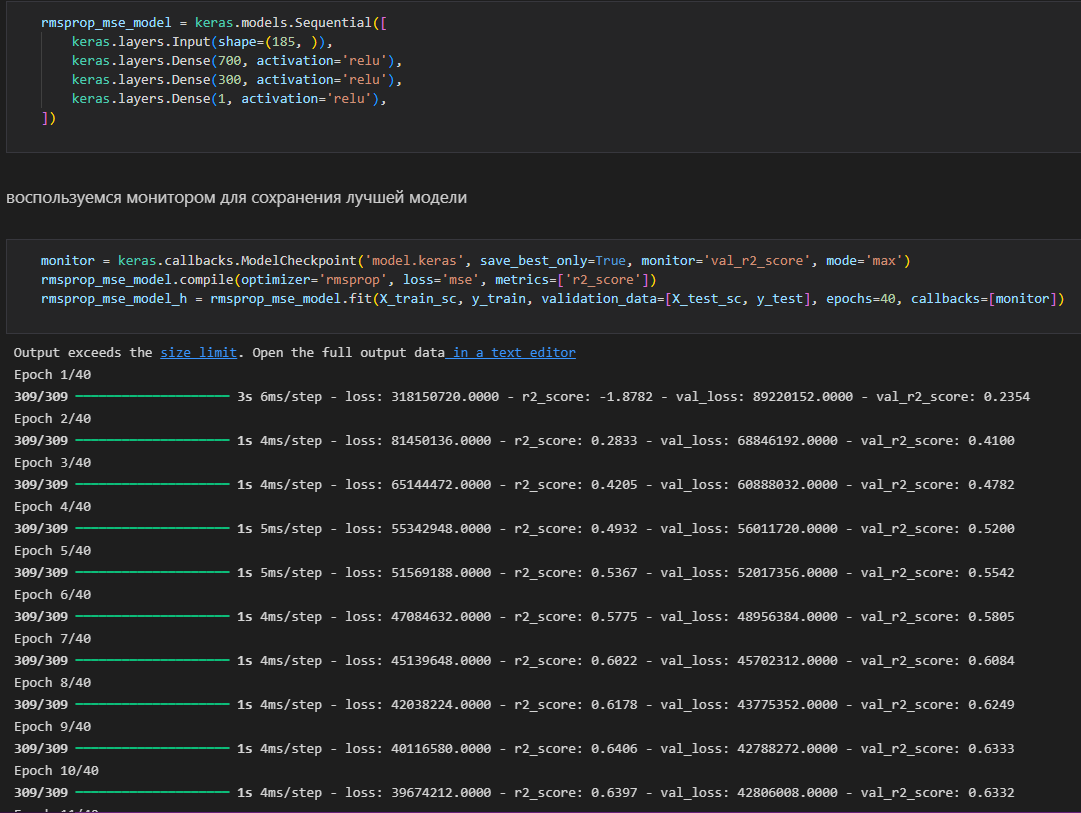


Рисунок 10 – Четвертая модель

Скорость обучения модели составляет примерно 4 секунды и потери минимально составляют 23647528

**3.1.** **Выбор способа оценки адекватности результатов обучения**

Оценка будет проходить с помощью функции r2\_score библиотеки scikit-learn. Оценим каждую модель отдельно (см. рис. 11,12,13,14)

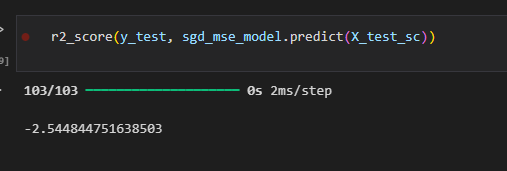


Рисунок 11 – Оценка первой модели

Оценка первой модели составляет -2.544844751638503

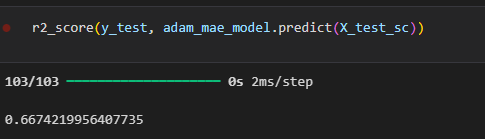


Рисунок 12 – Оценка второй модели

Оценка второй модели составляет 0.6674219956407735

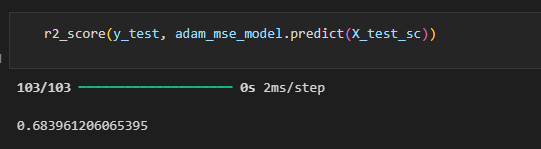


Рисунок 13 – Оценка третьей модели

Оценка третьей модели составляет 0.683961206065395

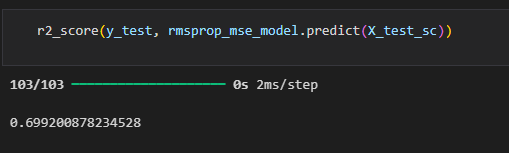


Рисунок 14 – Оценка четвертой модели

Оценка четвертой модели составляет 0.699200878234528

Для наглядного сравнения оценок каждой модели выведем график и обозначим результаты каждой модели (см. рис 14)

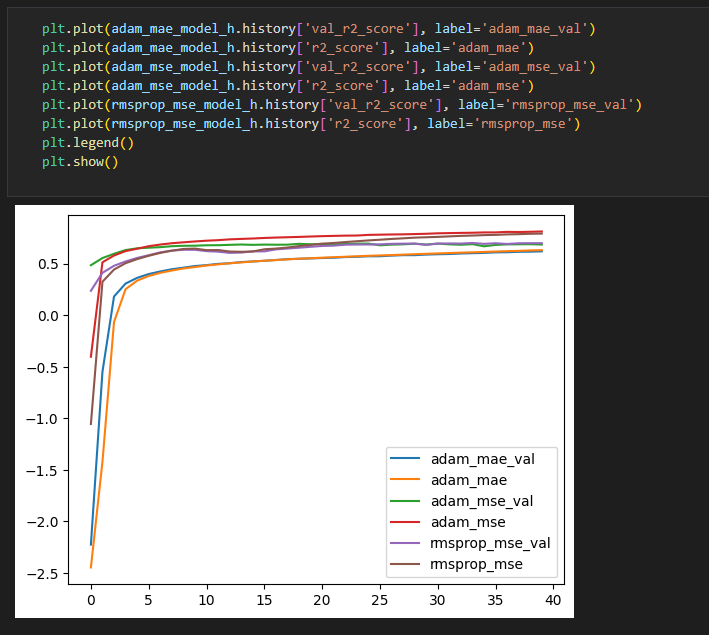
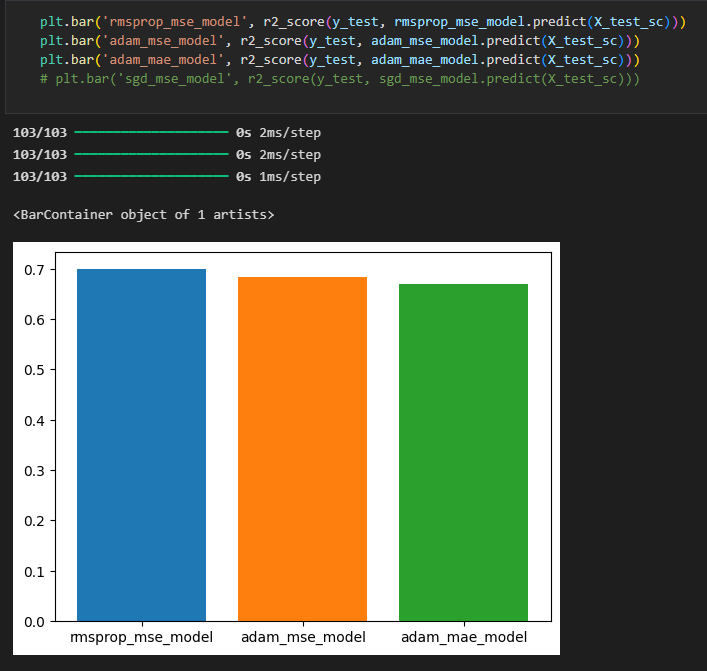


Рисунок 14 – График оценки

Исходя из графика можно сделать вывод , что все модели имеют хороший и приблизительный результат, но можно отметить что модель с оптимизатором adam и функцией потерь mse , а также модель с оптимизатором msprop и функцией потерь mse показали лучшие результаты.

* 1. **Анализ адекватности обучения**

Для сравнения всех моделей выведем график исходя из их оценок (см. рис. 15)

Рисунок 15 – сравнение моделей

Лучшей моделью оказалась последняя с оптимизатором rmsprop функцией потерь mse.

* 1. **Вывод по разделу**

|  |  |  |
| --- | --- | --- |
| Выводы | Сформированные компетенции | |
| Код компетенции | Содержание компетенции |
| Разработаны нейроны и нейронные блоки на языке программирования Python.  Создана нейронная сеть. Экспериментальным способом подобраны характеристики нейронной сети: определено количество слоев, число блоков в скрытых слоях, наличие или отсутствие обходных соединений, передаточные функции нейронов.  Собраны данные для обучения нейронной сети. Параметры обучения подобраны экспериментальным способом.  Произведено обучение нейронной сети на собранных данных. Проверена адекватность результатов обучения сети. | ПК-9 | Умение разрабатывать методы извлечения, анализа и обработки информации |
| ПК-10 | Способность применять математические методы моделирования процессов обработки информации с использованием средств интеллектуального анализа данных и машинного обучения |

**ЗАКЛЮЧЕНИЕ**

В рамках практики была проведена самостоятельная работа на всех этапах решения учебных бизнес-задач, поставленных университетом. Это включало в себя изучение технического и программного обеспечения, а также освоение технологий выполнения задач и процессов.

За время прохождения производственной (эксплуатационной) практики были получены следующие профессиональные навыки:

- Установка отдельных компонентов прикладного программного обеспечения.

- Математическое моделирование нейронной сети и визуализация полученных данных.

- Разработка и обучение нейронных сетей.

- Документирование процессов создания нейронных сетей в соответствии со стандартами ГОСТ 34.602-89 и ГОСТ 19.201-78.

- Анализ, подбор, подготовка и нормализация данных для обучения нейронных сетей.

- В результате производственной практики (эксплуатационной практики) сформировались следующие навыки:

- Использование математических моделей при разработке программных решений.

- Автоматизация решения типовых задач и интеграция разнородных данных в информационных системах.

- Разработка и адаптация прикладного программного обеспечения для выполнения задач в конкретной области.

- Оценка и уменьшение потерь в процессе тренировки нейронной сети.

- Проверка результатов обучения нейронной сети на соответствие ожиданиям.

- Анализ собранных данных и подготовка отчётов по практике.

В результате цель практики была достигнута за счёт закрепления теоретических знаний, получения практических навыков и опыта работы в данной сфере, а также изучения профессиональных вопросов, что способствовало полноценному формированию знаний и опыта профессиональной деятельности.

**СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ И ЛИТЕРАТУРЫ**

1. Habr: сайт. – Москва. – Обновляется в течении суток – URL: <https://habr.com/ru/articles/312450/> (дата обращения 21.05.24) – Текст. – Изображение: электронные.
2. Yandex: сайт. – Санкт-Петербург. – Обновляется в течении суток – URL: https://practicum.yandex.ru/blog/chto-takoe-neyronnye-seti/ (дата обращения 22.05.24) – Текст. – Изображение: электронные.
3. Sky.pro: сайт. – Москва. – Обновляется в течении суток – URL: https://sky.pro/media/neyronnye-seti/ (дата обращения 14.05.24) – Текст. – Изображение: электронные.
4. Neiros: сайт. – Воронеж. – Обновляется в течении суток – URL: https://neiros.ru/blog/automation/neyronnye-seti-printsip-raboty-perspektivy-i-159-sovremennykh-neyronok/ (дата обращения 01.05.24) – Текст. – Изображение: электронные.
5. Гафаров, Ф.М. Искусственные нейронные сети и приложения: учеб. пособие / Ф.М. Гафаров, А.Ф. Галимянов. – Казань: Изд-во Казан. ун-та, 2018. – 121 с. – Режим доступа: - URL: <https://kpfu.ru/staff_files/F1493580427/NejronGafGal.pdf> (дата обращения 12.05.24). -Текст: электронный.