МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Московский государственный технический университет   
имени Н.Э. Баумана

(национальный исследовательский университет)»

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

**по курсу**

«Data Science»

Слушатель Головин Евгений Владимирович

Москва, 2022

**Содержание**

1. Введение ………………………………………………………….. ..3
2. Основная часть Решение задач квалификационной работы……..4
   1. Разведочный анализ данных…………………………………....4
   2. Задача регрессии методами машинного обучения………......14
   3. Применение нейронной сети для рекомендации соотношения

матрица-наполнитель………………………………………….21

* 1. Разработка приложения с интерфейсом командной строки...26

1. Заключение. Результаты и выводы...……………………………...27

Библиографический список………………………………………..28

1. **Введение**

Композиционные материалы - материалы, в которых имеет место сочетание двух или более химически разнородных компонентов с четкой границей раздела между ними. Это неоднородные по химическому составу и структуре материалы. Структура композиционных материалов представляет собой матрицу (основной компонент), содержащую в своем объеме или армирующие элементы, часто называемые наполнителем. Матрица и наполнитель разделены границей (поверхностью) раздела. Наполнитель равномерно распределен в матрице и имеет заданную пространственную ориентацию.

Композиционные материалы характеризуются совокупностью свойств, не присущих каждому в отдельности взятому компоненту. За счет выбора армирующих элементов, варьирования их объемной доли в матричном материале, а также размеров, формы, ориентации и прочности связи по границе «матрица-наполнитель», свойства композиционных материалов можно регулировать в значительных пределах.

Композиционные материалы используются: в авиационной, ракетной и космической технике, в металлургии, в горнорудной промышленности, в химической промышленности, в автомобильной промышленности, в сельскохозяйственном машиностроении, в электротехнической промышленности, в ядерной технике, в машиностроительной отрасли, в сварочной технике, в судостроительной промышленности, в медицинской промышленности, в строительстве, в бытовой технике.

Широкое распространение и высокая потребность в новых материалах обуславливают актуальность темы данной работы.

**1.1 цели и задачи квалификационной работы**

Цель – продемонстрировать знания и навыки в области Data Science, машинного обучения, баз данных, нейросетей и больших чисел, которые были сформированы в период прохождения обучения на квалификационном курсе.

Для достижения цели следует решить следующие задачи:

1. Провести разведочный анализ предложенных данных и провести предобработку данных.
2. Обучить несколько моделей для прогноза модуля упругости при растяжении и прочности при растяжении. Оценить точность модели на тренировочном и тестовом датасете.
3. Написать нейронную сеть, которая будет рекомендовать соотношение матрица-наполнитель. Оценить точность прогноза на тренировочном и тестовом датасете.
4. Разработать приложение с интерфейсом командной строки, которое будет выдавать один или два прогноза.
5. **Основная часть. Решение задач квалификационной работы**

**2.1** **Разведочный анализ данных**

Скрипт на языке Python 3 начинаем с импорта библиотек, рисунок 1:



Рисунок 1 – импорт библиотек.

Далее укажем глобальные настройки и константы, рисунок 2:

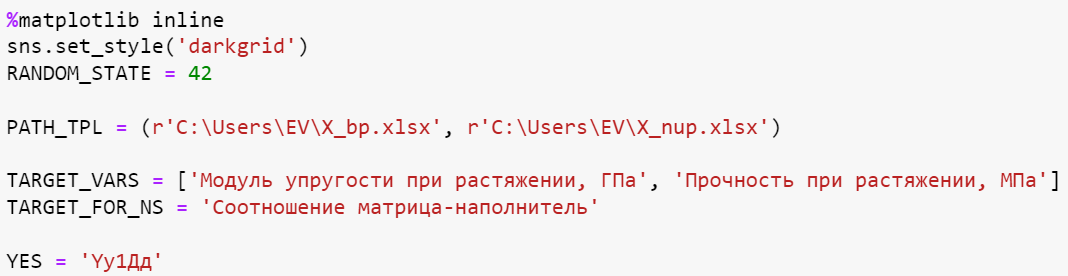
****

Рисунок 2 - глобальные настройки и константы

Для разведочного анализа потребуется ряд функций, рисунки 3 – 5.



Рисунок 3 – функции для разведочного анализа данных

****

****

Рисунок 4 – функции для разведочного анализа данных

****

Рисунок 5 – функции для разведочного анализа данных

В функции EDA\_search\_dialog (рисунок 5) реализован диалог с пользователем, позволяющий выводить требуемые аналитические построения с учетом времени выполнения функции.

Основное тело скрипта для решения первой задачи реализовано в блоке вызова функций, рисунок 6.

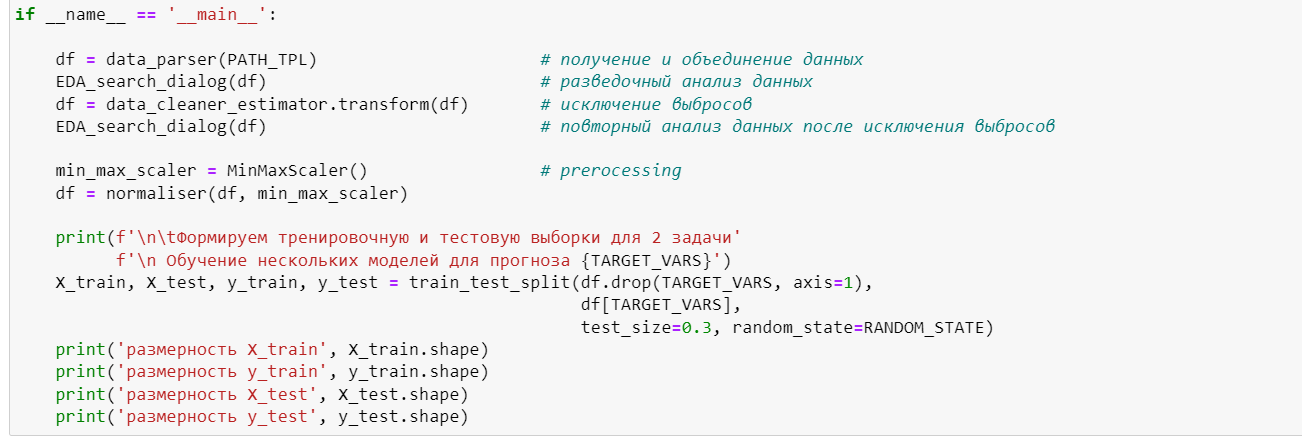


Рисунок 6 – вызов функций формирования датасета его анализа и очистки

Указанный скрипт позволяет начать работу с набором данных, переведя данные в формат Pandas DataFrame, провести слияние двух частей набора данных, вывести описательную статистику в виде текста, таблиц и диаграмм, рисунок 7.

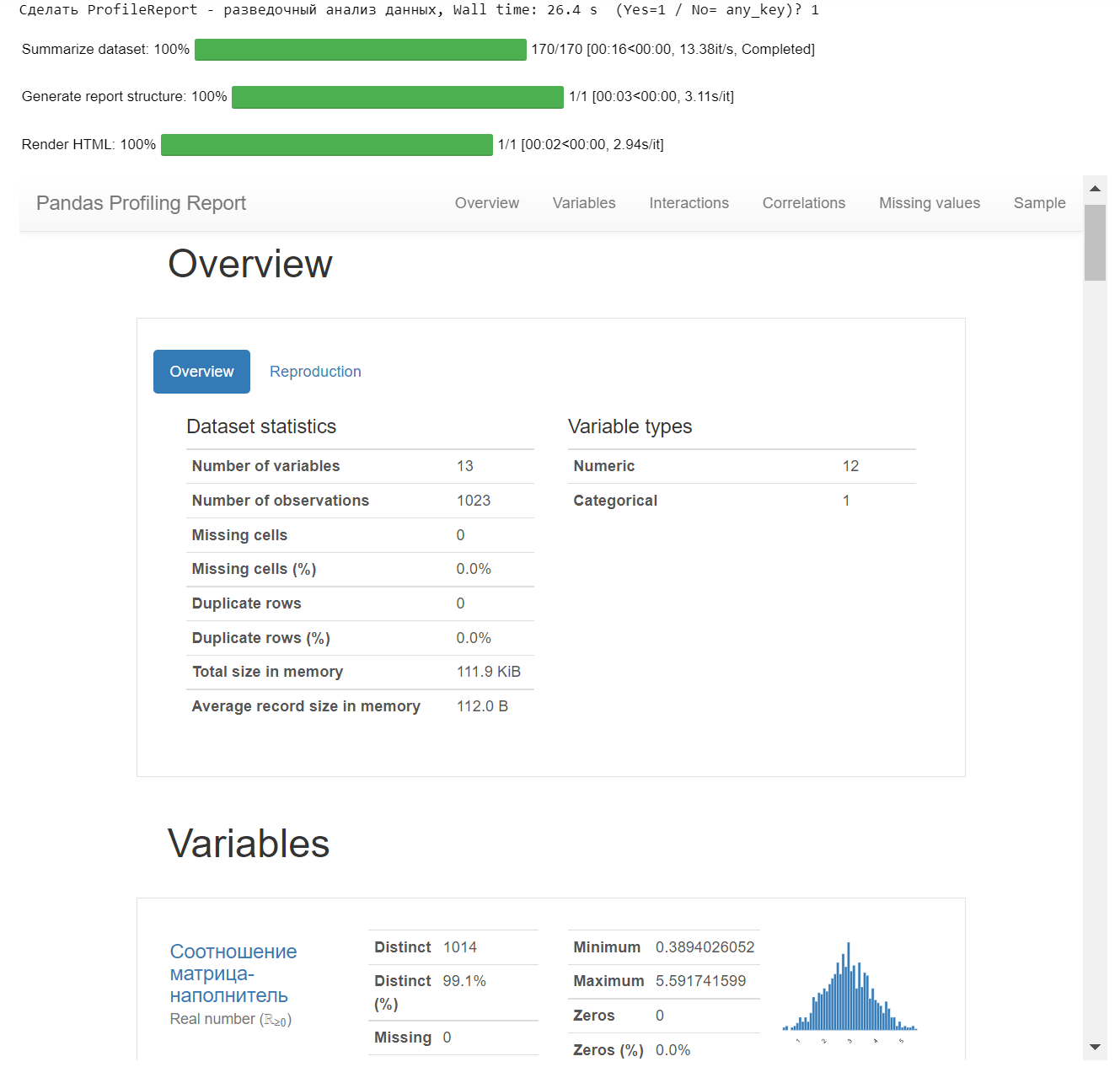


Рисунок 7 – результаты разведочного анализа данных

Этот этап исследования позволяет получить карту плотности распределений, рисунок 8, гистограммы распределений и «ящики с усами», рисунок 9.

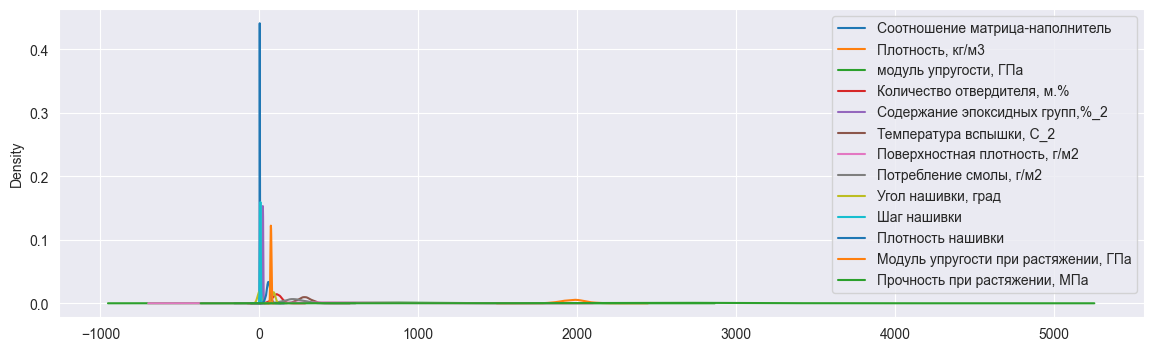
**

Рисунок 8 - карта плотности распределений KDE

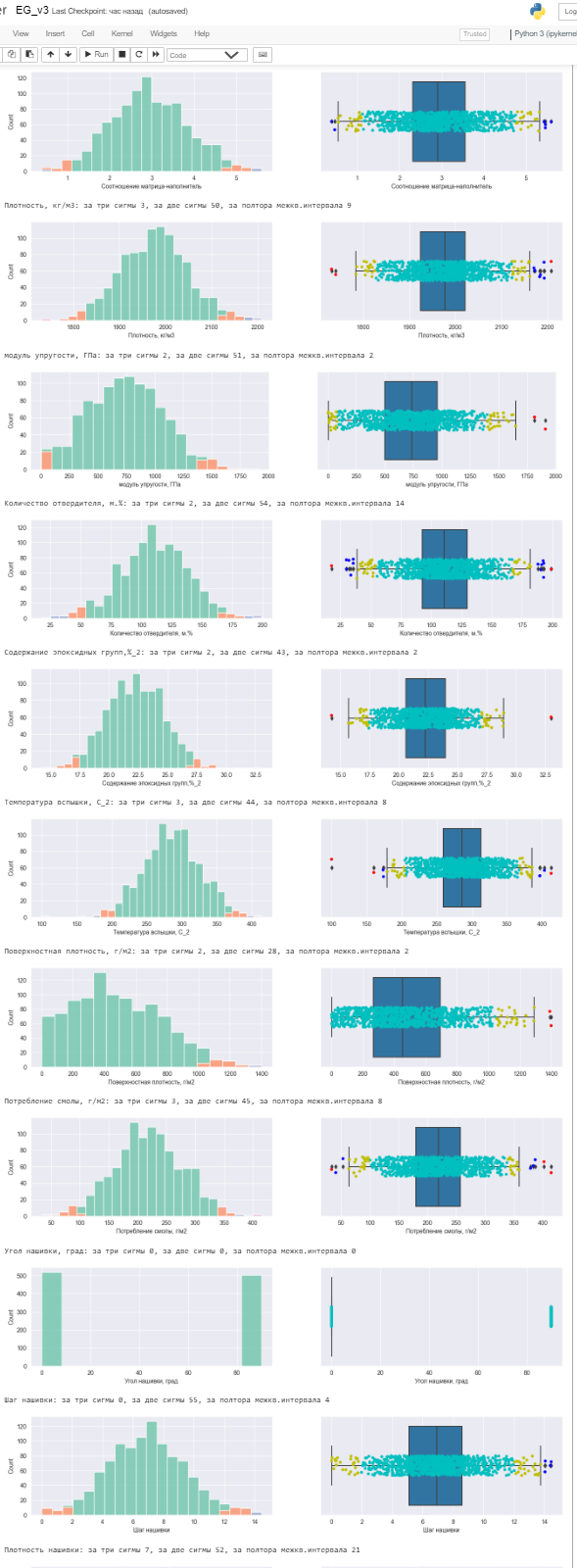
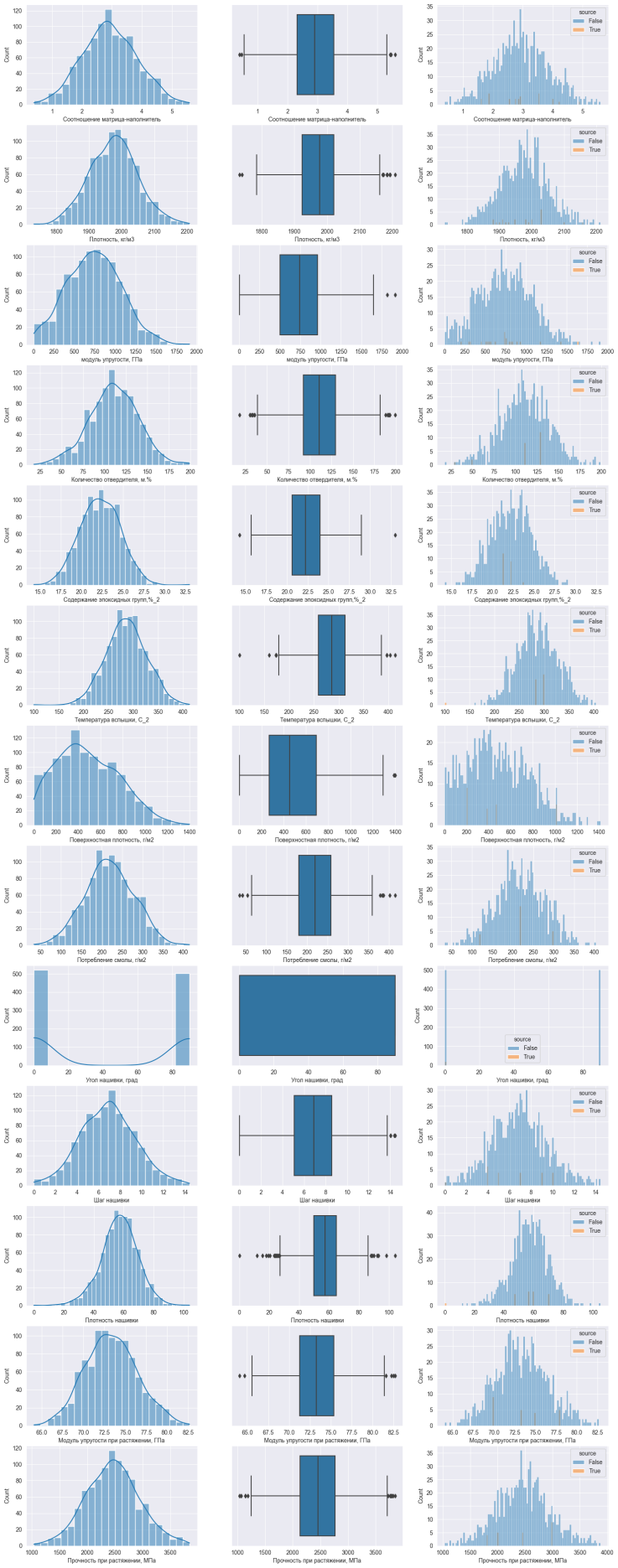
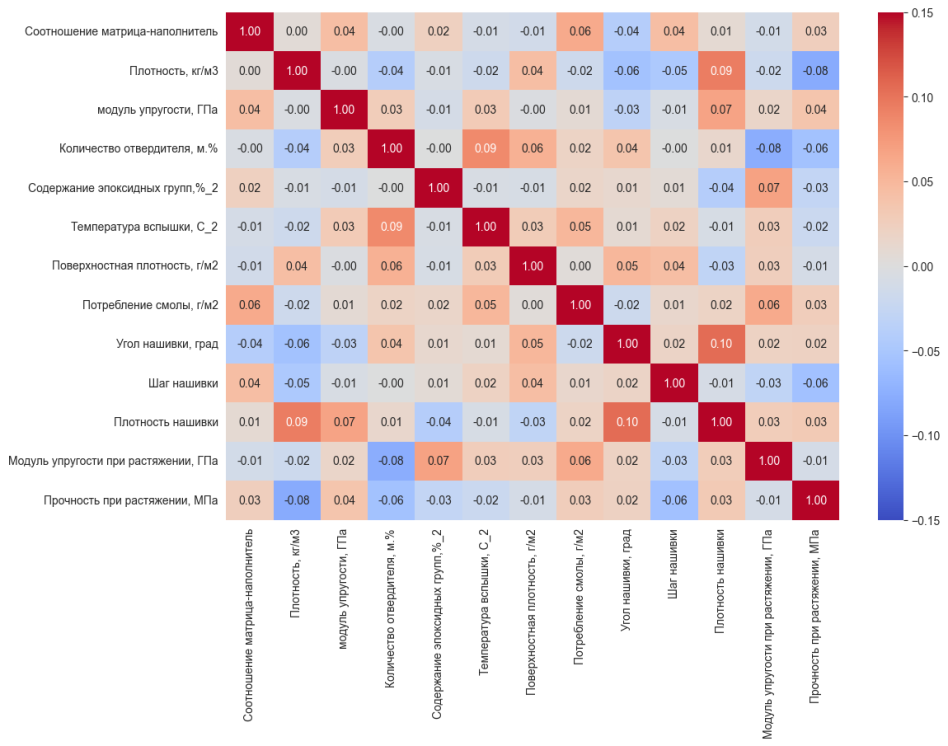


Рисунок 9 - Гистограммы распределения переменных, «ящики с усами», визуализация выбросов, рассчитанная 3-мя способами.

Скрипт выдает также таблицы с основными параметрами исходных и целевых переменных: среднее, медиана, число записей, число уникальных значений, считает повторы в столбцах, рисунок 10.

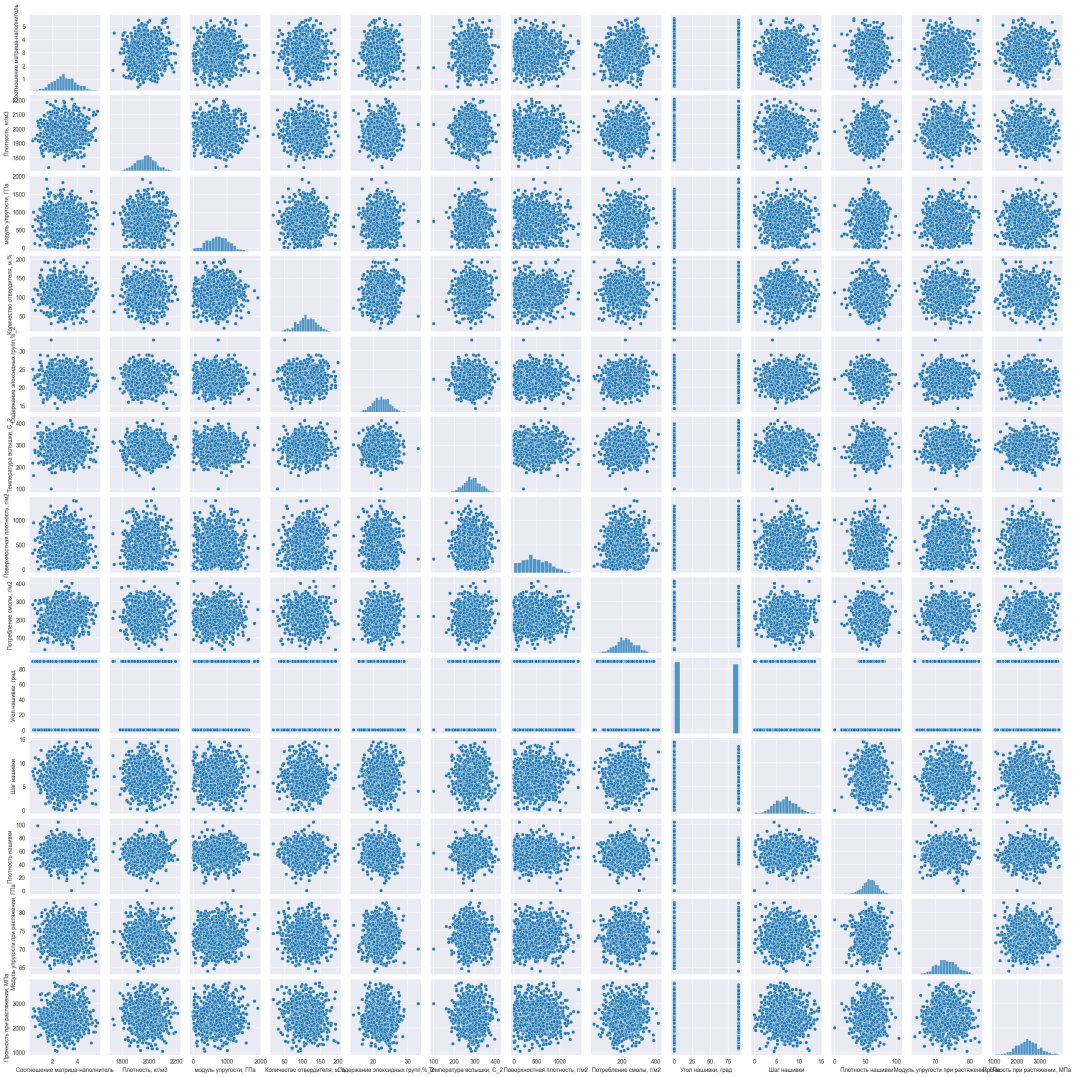
**­**

****

****

Рисунок 10 – Описательная статистика

Корелляционная матрица позволяет визуально оценить наличие явных зависимостей между переменными, рисунок 11

****

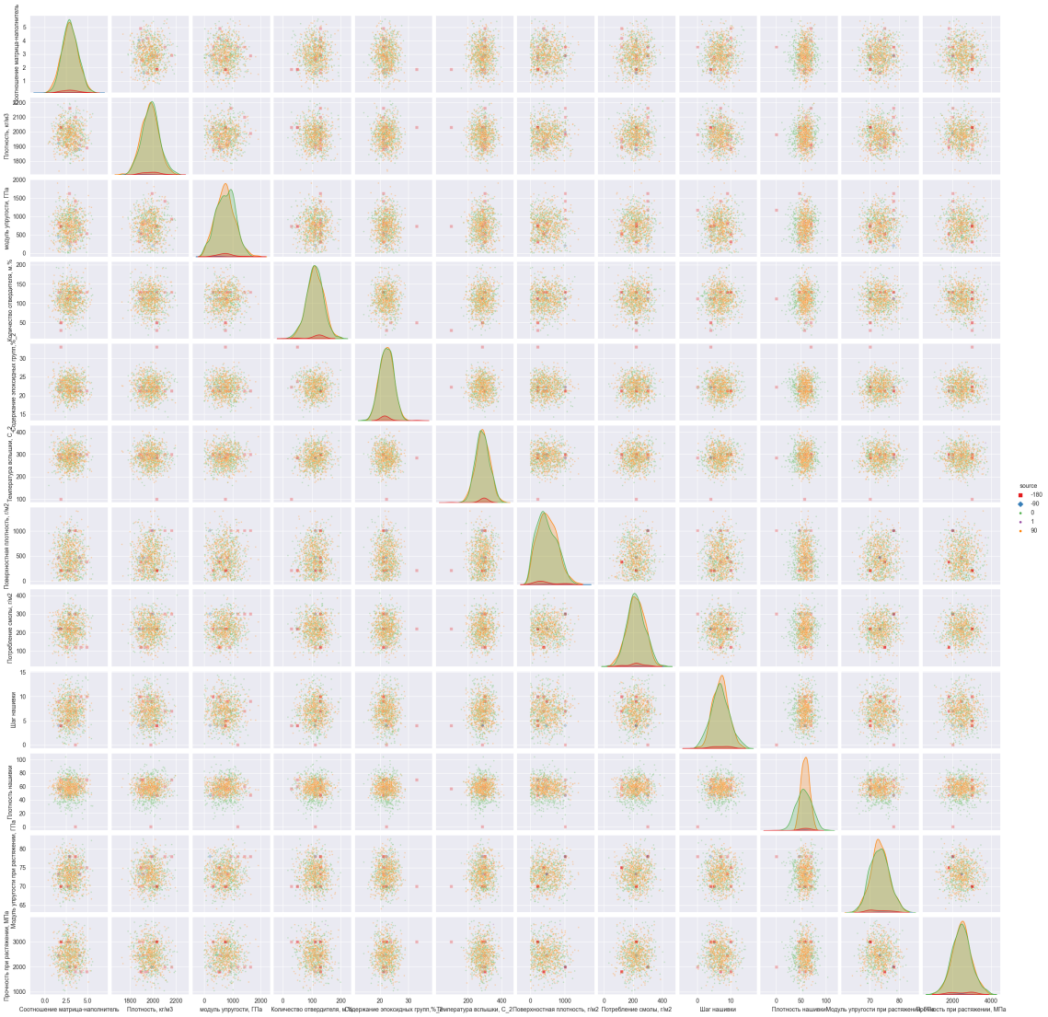
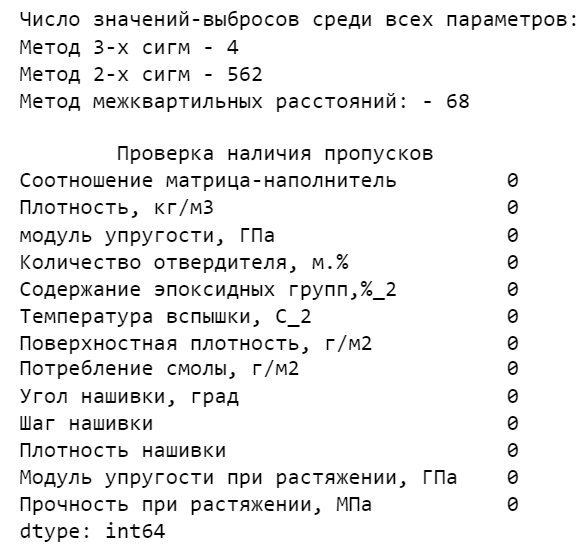
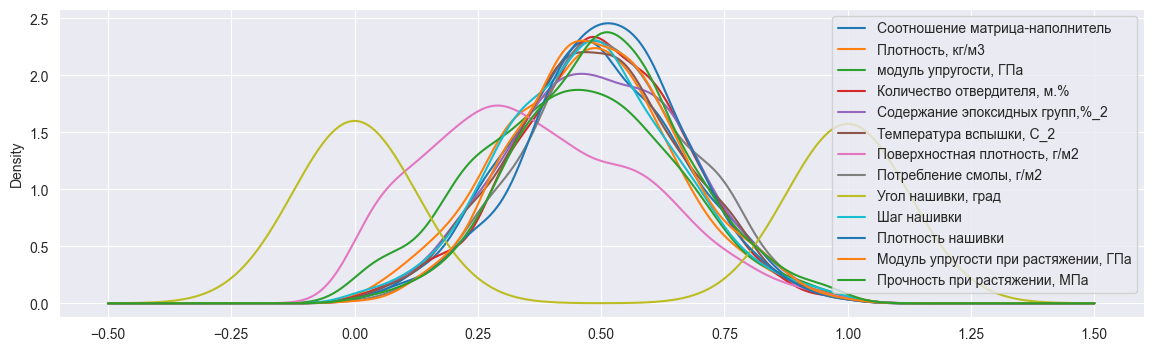


рисунок 11 – корелляционные мтрицы

****

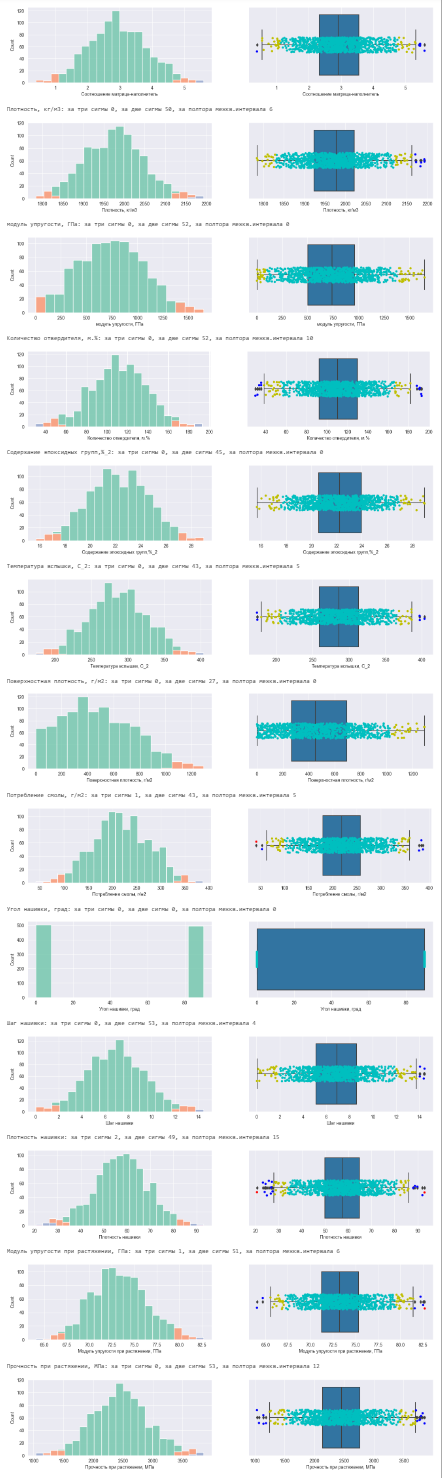
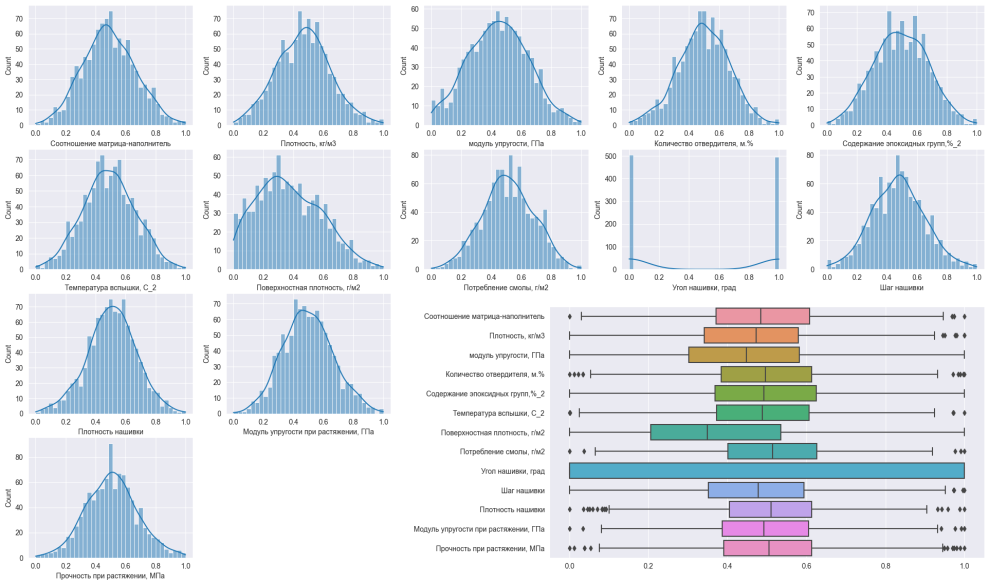
****

Рисунок 12 – результаты очистки и нормализации данных

Нормализация данных проводилась методом MinMaxScaller, поскольку данные представляют собой измерения физических величин, имеющих положительное значение в определенном интервале.

В таблице 1 приведены показатели описательной статистики после нормализации данных.

Таблица 1 – описательная статистика нормализованного набора данных

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | count | mean | std | min | 25% | 50% | 75% | max |
| Соотношение матрица-наполнитель | 1000.00 | 0.49 | 0.17 | 0.00 | 0.37 | 0.48 | 0.61 | 1.00 |
| Плотность, кг/м3 | 1000.00 | 0.47 | 0.18 | 0.00 | 0.34 | 0.47 | 0.58 | 1.00 |
| модуль упругости, ГПа | 1000.00 | 0.45 | 0.20 | 0.00 | 0.30 | 0.45 | 0.58 | 1.00 |
| Количество отвердителя, м.% | 1000.00 | 0.50 | 0.17 | 0.00 | 0.38 | 0.50 | 0.61 | 1.00 |
| Содержание эпоксидных групп,%\_2 | 1000.00 | 0.49 | 0.18 | 0.00 | 0.37 | 0.49 | 0.62 | 1.00 |
| Температура вспышки, С\_2 | 1000.00 | 0.49 | 0.17 | 0.00 | 0.37 | 0.49 | 0.61 | 1.00 |
| Поверхностная плотность, г/м2 | 1000.00 | 0.37 | 0.22 | 0.00 | 0.21 | 0.35 | 0.54 | 1.00 |
| Потребление смолы, г/м2 | 1000.00 | 0.51 | 0.17 | 0.00 | 0.40 | 0.51 | 0.63 | 1.00 |
| Угол нашивки, град | 1000.00 | 0.50 | 0.50 | 0.00 | 0.00 | 0.00 | 1.00 | 1.00 |
| Шаг нашивки | 1000.00 | 0.48 | 0.18 | 0.00 | 0.35 | 0.48 | 0.59 | 1.00 |
| Плотность нашивки | 1000.00 | 0.51 | 0.16 | 0.00 | 0.41 | 0.51 | 0.61 | 1.00 |
| Модуль упругости при растяжении, ГПа | 1000.00 | 0.50 | 0.17 | 0.00 | 0.39 | 0.49 | 0.61 | 1.00 |
| Прочность при растяжении, МПа | 1000.00 | 0.51 | 0.17 | 0.00 | 0.39 | 0.50 | 0.61 | 1.00 |

Как видим, повторы по отдельным переменным есть только в первых 23 строках взятых из X\_bp.xlsx и в первых 40 строках из X\_nup.xlsx, при этом нет дубликатов по всем параметрам. Удаления из датасета не требуют.

Результаты первого этапа – разведочного анализа – позволяют нам применить щадящую очистку от выбросов и нормализацию, после чего можно приступить к формированию и обучению моделей машинного обучения.

**2.2 Задача регрессии методами машинного обучения**

Для решения этой задачи понадобятся следующие функции, рисунок 14.



Рисунок 15 – функции для машинного обучения и анализа моделей.

Формируем тренировочную и тестовую выборки для 2 задачи.

размерность X\_train (700, 11)

размерность y\_train (700, 2)

размерность X\_test (300, 11)

размерность y\_test (300, 2)

Основные модели библиотеки ScikitLearn, способные решать задачи   
регресссии, занесем в словарь, а также составим словарь метрик, которые   
будем применять для оценки эффективности гиперпараметров моделей и   
выбора лучшей модели, рисунок 16.

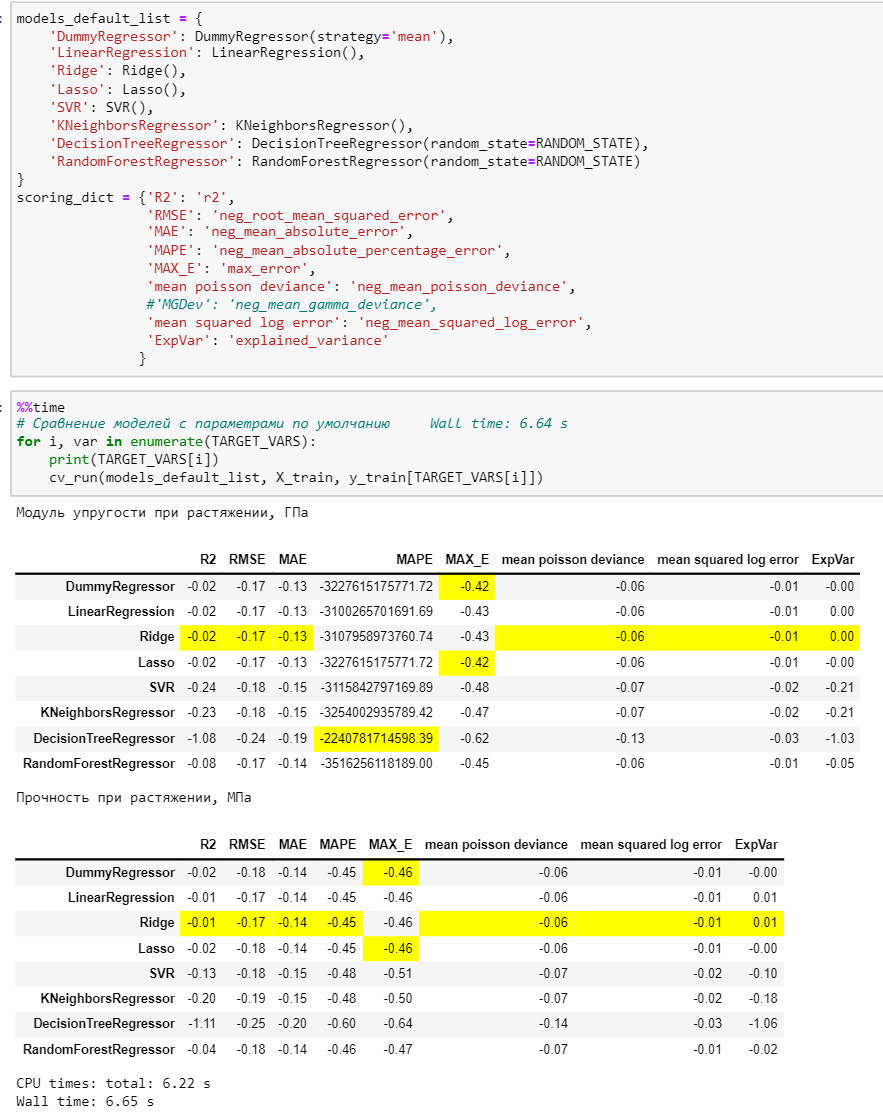


Рисунок 16 – словари моделей и метрик, метрики моделей с параметрами по   
умолчанию.

Без оптимальных гиперпараметров алгоритмы выдают отрицательные   
показатели по всем метрикам ошибок, что говорит о том, что их прогноз хуже   
применения среднего значения во всех случаях.

Кросс-валидация позволяет нам подобрать оптимальные параметры для каждого из алгоритмов после обучения на тренировочной части набоа данных   
для каждой целевой переменной из двух (модуль упругости при растяжении и   
прочность при растяжении).

Импользуем поиск параметров по сетке для каждого алгоритма,   
наблюдаем первые 3 ранга лучших наборов параметров. Сохраняем лучшие   
модели, рисунок 17.

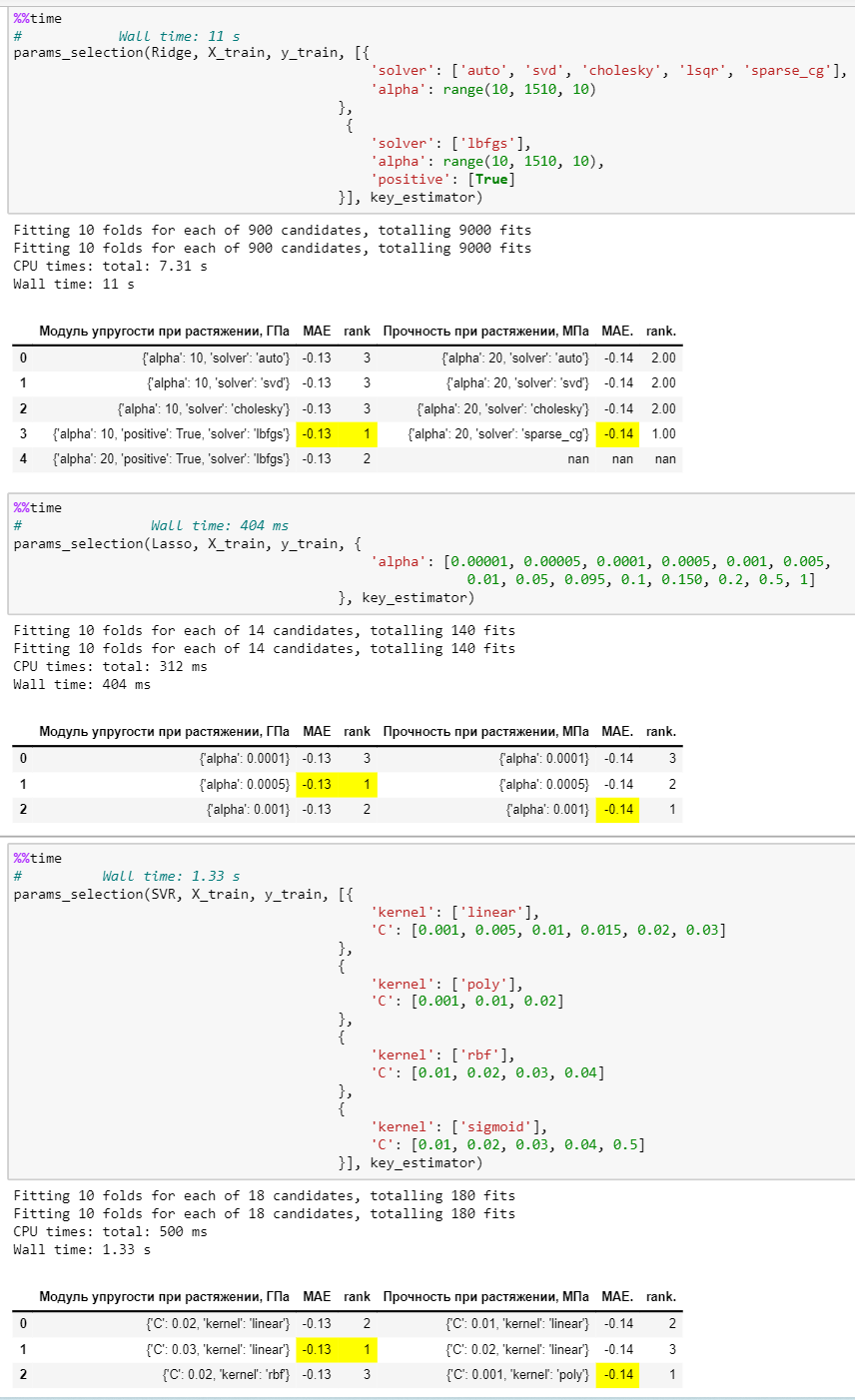


Рисунок 17 - Модели Ridge(), Lasso(), SVR()

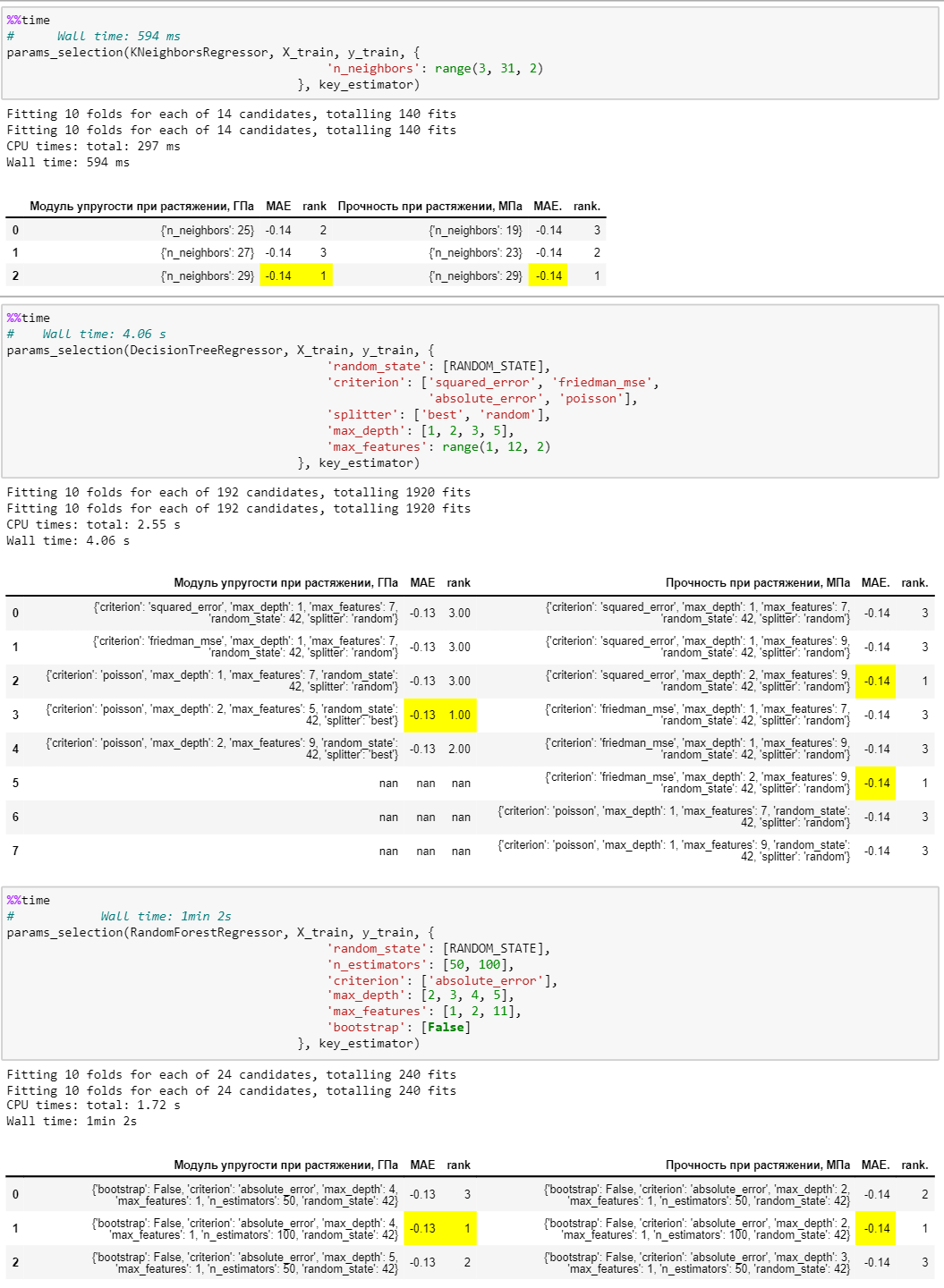


Рисунок 18 – Модели KNeighborsRegressor(), DecisionTreeRegressor(),   
RandomForestRegressor().

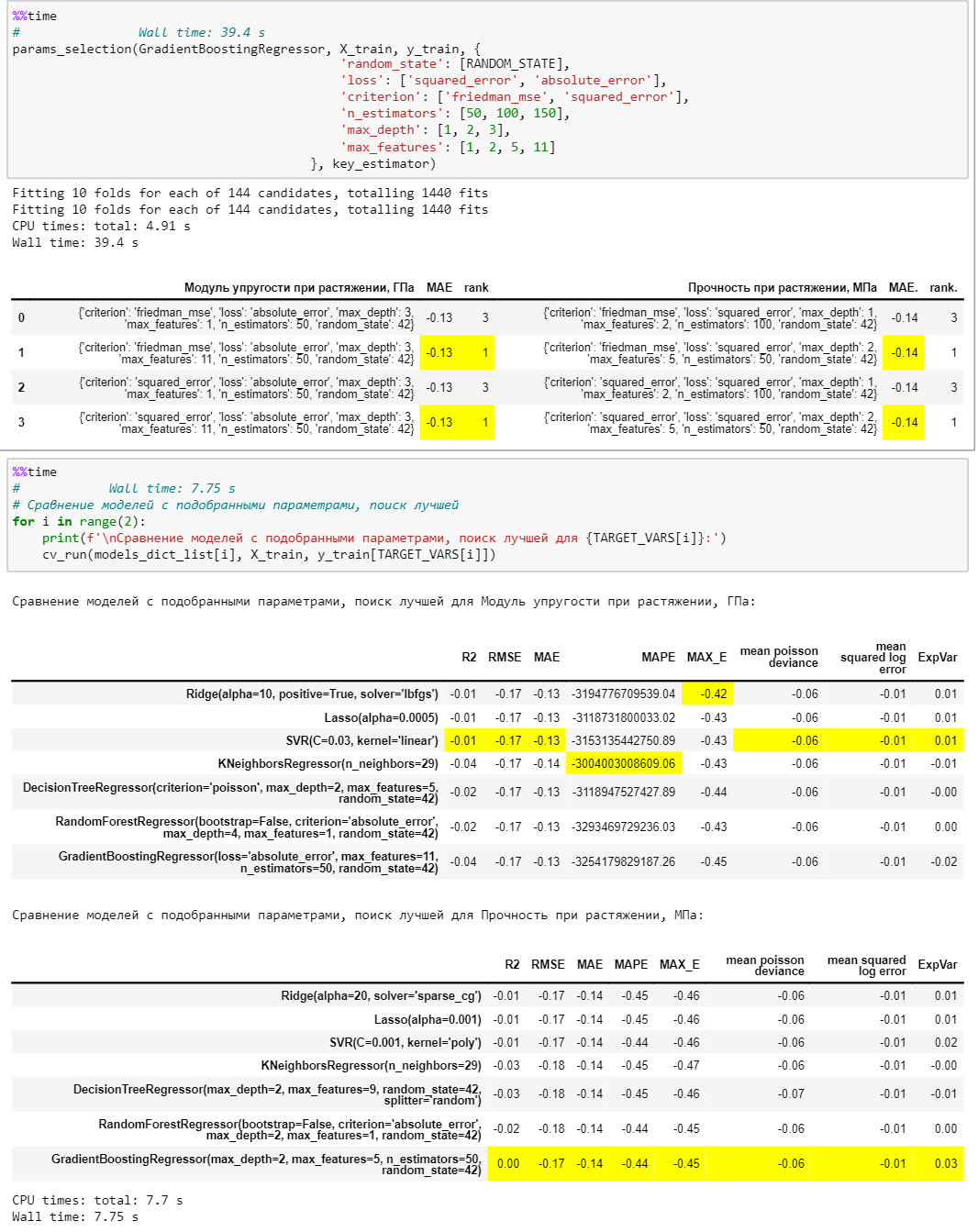


Рисунок 18 – Модель GradientBoostingRegressor() и общее сравнение моделей

Чтобы помимо числовых значений визуально оценить эффективность моделей добавим ещё пару вспомогательных функций, рисунок 19.



Рисунок 19

По каждому алгоритму с подобранными по сетке гиперпараметрами мы   
получаем графики соответствия прогноза и тестовых данных и числовые показатели различных метрик ошибок, рисунок 20.

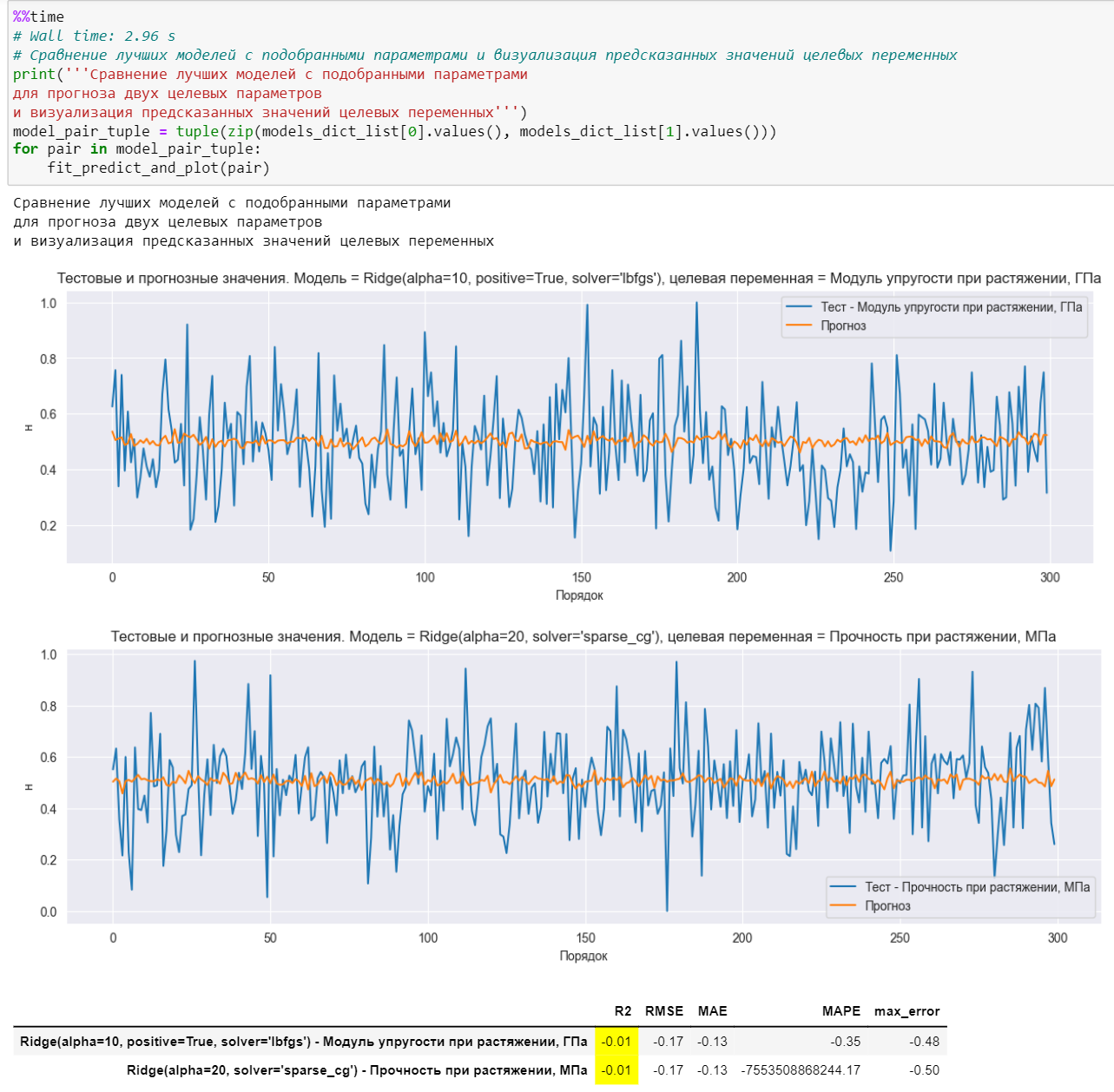


Рисунок 20 - графики прогноза и тестовых данных и показатели метрик ошибок

**2.3 Применение нейронной сети для рекомендации соотношения  
матрица-наполнитель.**

Для формирования рекомендательной нейронной сети также   
потребуется ряд функций, рисунок 21.



Рисунок 21 – функции оценки эффективности для нейросетей

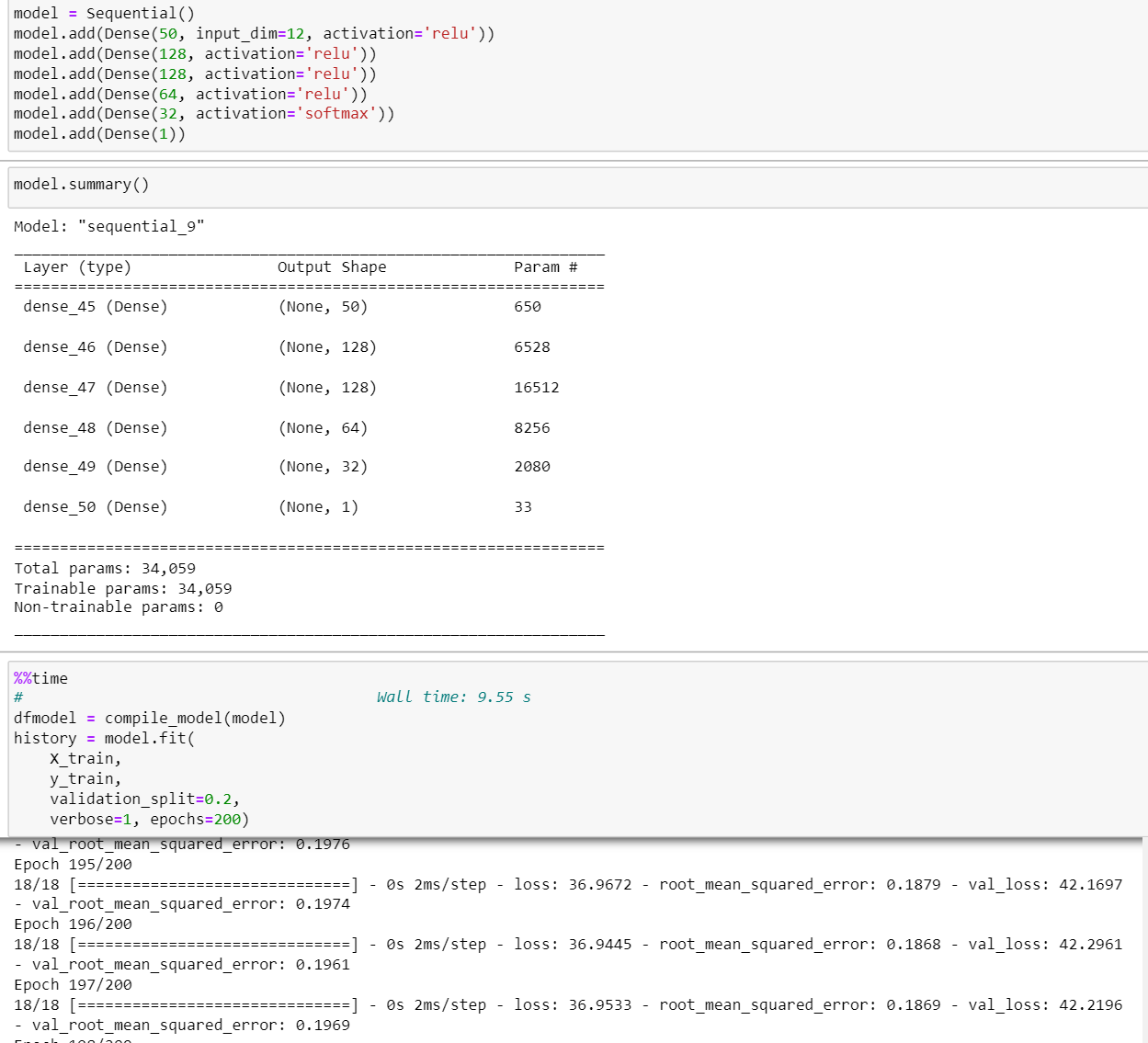


Рисунок 22

И предложим собственно сами варианты нейронных сетей. В частности, для решения этой задачи можно применить многослойный персептрон.   
Попробуем несколько персипторнов с различным числом скрытых слоев и   
числом нейронов на скрытых слоях, а также с дропаут слоями и с применением ранней остановки обучения сети, рисунок 22.

Эффективность будем оценивать по графику снижения ошибок   
RMSE (root\_mean\_squared\_error) и MAPE (mean\_absolute\_percentage\_error) и по наложению графиков прогноза и тестовой целевой переменной,   
рисунки 23 – 25, таблица 2.

Таблица 2. Сравнение предсказаний базовой модели и лучшей модели на тестовом множестве

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | R2 | RMSE | MAE | MAPE | max\_error |
| DummyRegressor | -0.00 | -0.18 | -0.14 | -7371206150046.71 | -0.49 |
| model | -0.19 | -0.19 | -0.15 | -6129352749438.68 | -0.56 |
| model\_2 | -0.60 | -0.22 | -0.18 | -6449446371983.82 | -0.62 |
| MLPRegressor | -0.01 | -0.18 | -0.14 | -7521324657610.99 | -0.50 |

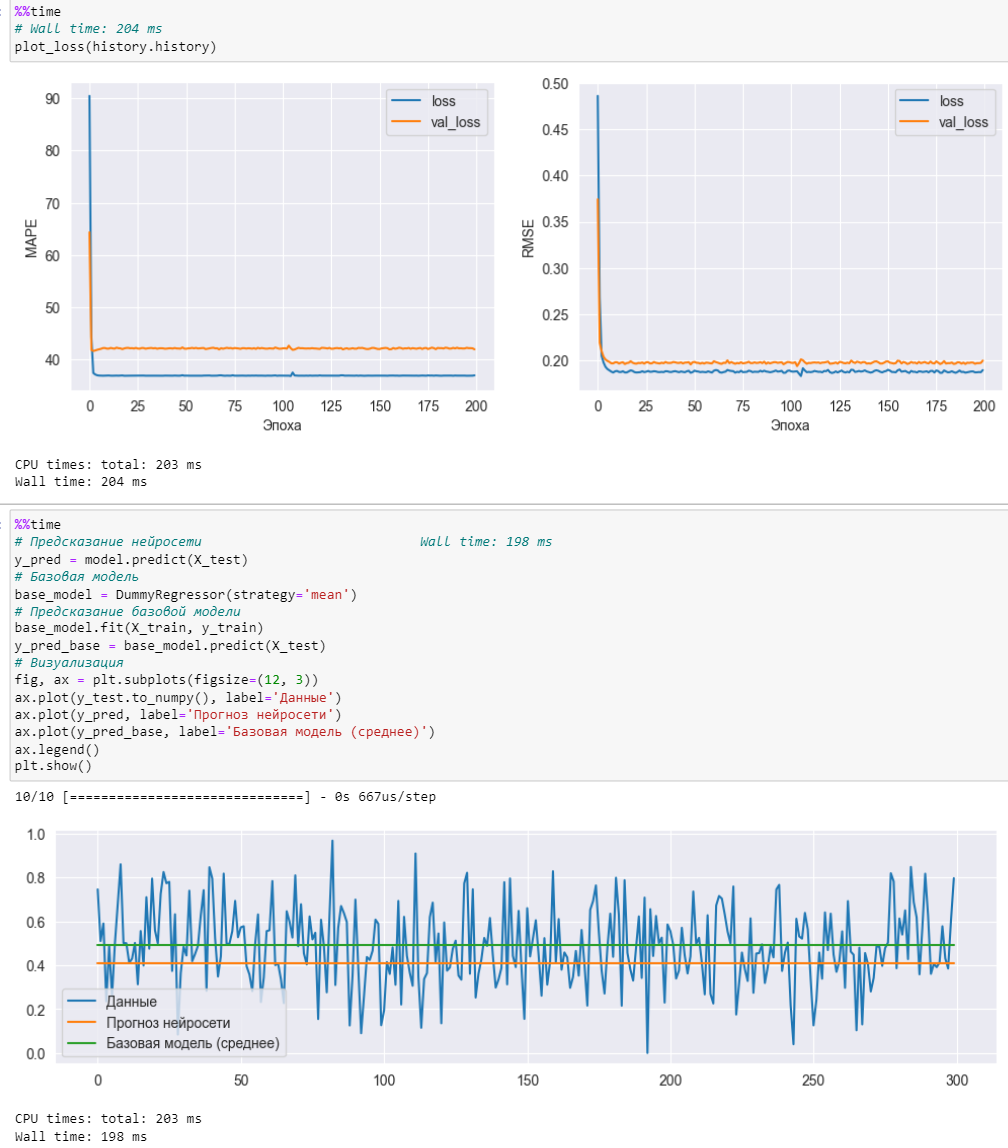


Рисунок 23 – Оценка эффективности многослойного персептрона

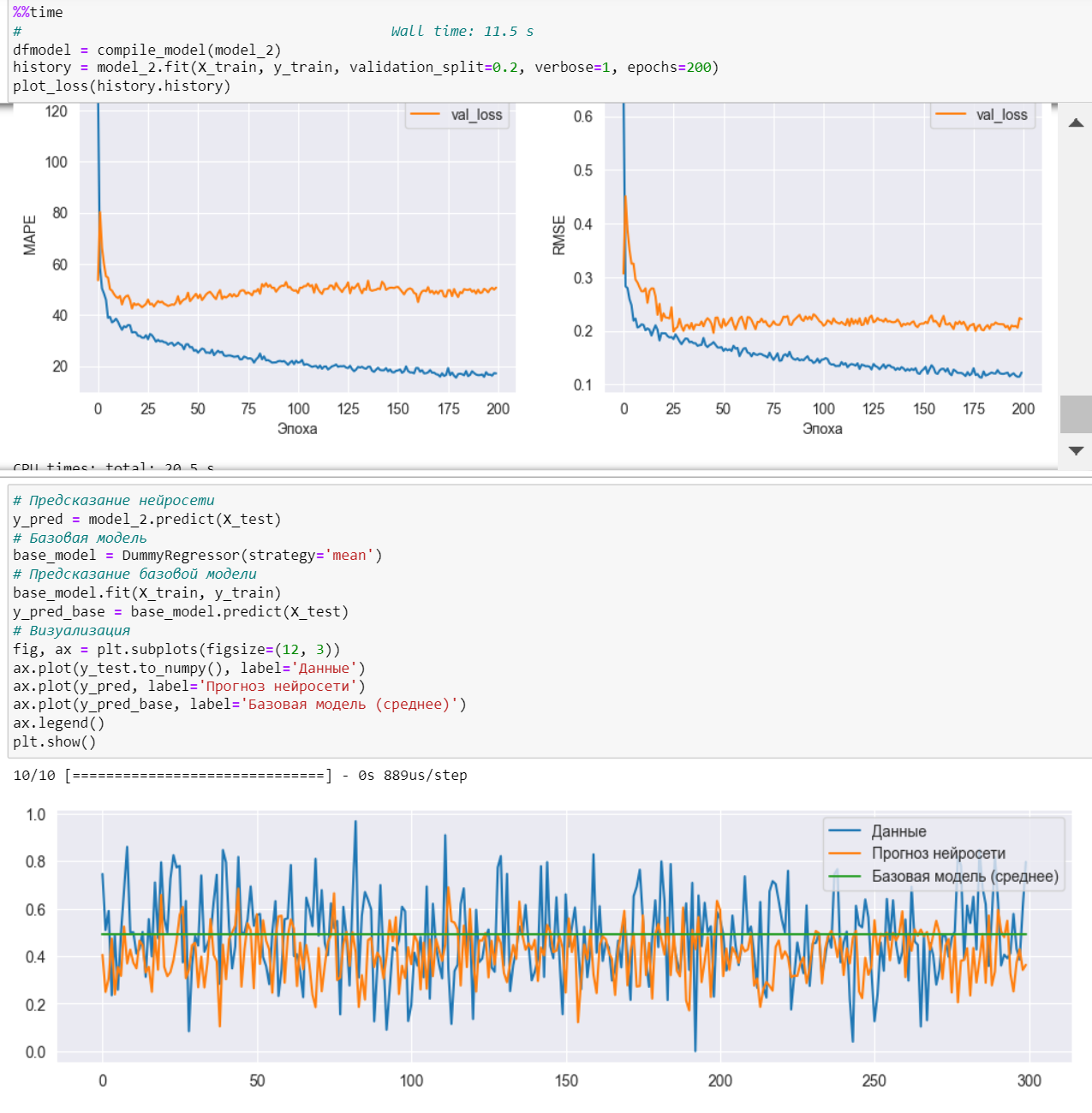


Рисунок 24 – Оценка эффективности многослойного персептрона



Рисунок 25 – Оценка эффективности многослойного персептрона

**2.4. Разработка приложения с интерфейсом командной строки**

Оценка моделей показала, что их применение в производстве нецелесообразно. Тем не менее функционал приложения и универсальность алгоритмов позволят быстрее внедрить рабочую модель, если она будет создана на этих или на дополненных данных.

В приложении необходимо реализовать следующие функции:

* выбор целевой переменной для предсказания;
* ввод входных параметров;
* проверка введенных параметров;
* загрузка сохраненной модели, получение и отображение прогноза выходных параметров.

Веб-приложение разработано с помощью языка Python, фреймворка Flask и шаблонизатора Jinja.

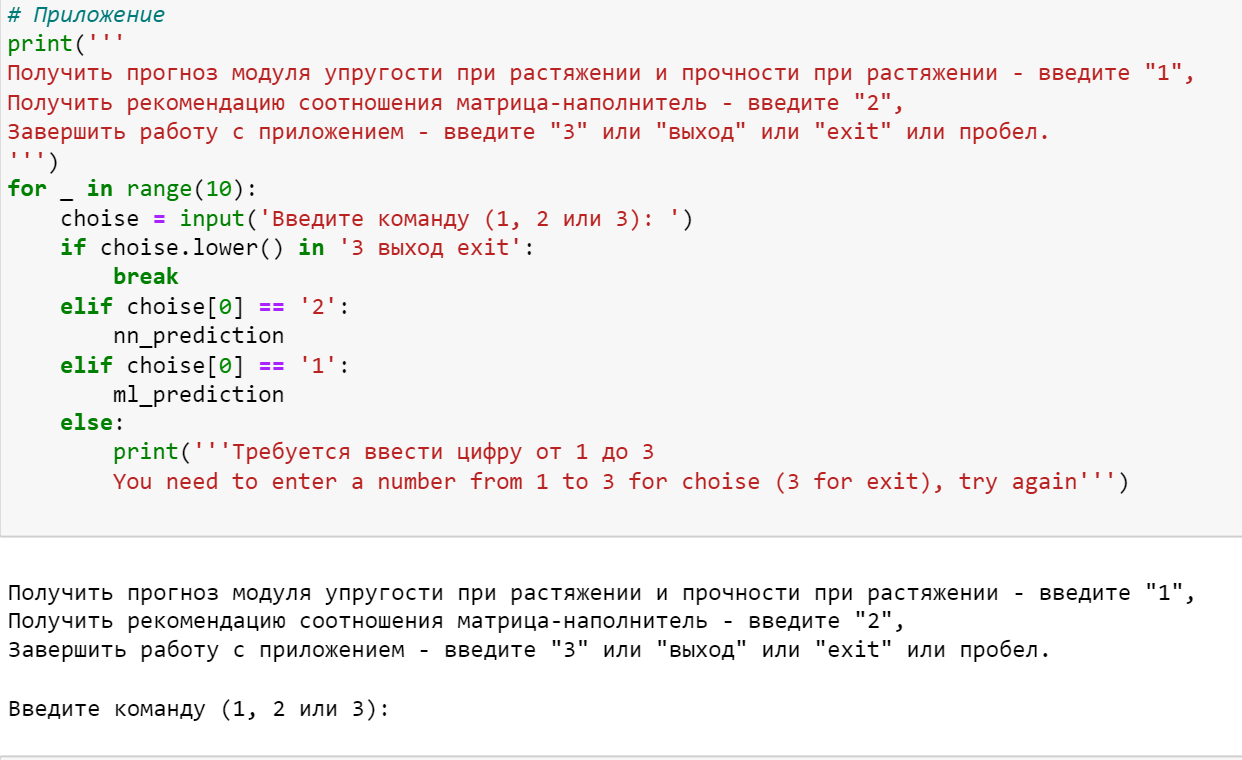


Рисунок 26 – скрипт приложения на Python-3

1. **Заключение. Результаты и выводы**

Стоимость производства композитного материала высока. Зная характеристики компонентов, невозможно рассчитать свойства композита. Значит для получения заданных свойств требуется большое количество испытаний различных комбинаций. Сократить время и затраты на создание определенного материала могла бы помочь система поддержки производственных решений, построенная на принципах машинного обучения.

Таким образом, вышеизложенные материалы демонстрируют требуемый объем знаний и навыков в области Data Science, машинного обучения, баз данных, нейросетей и больших чисел, которые были сформированы в период прохождения обучения на квалификационном курсе. Цель выпускной квалификационной работы достигнута в полном объеме.

В рамках данной работы решены следующие задачи:

1. Проведен разведочный анализ данных и предобработка данных (очистка от выбросов, нормализация, проверка пропусков и дубликатов).
2. Обучены 28 моделей для прогноза модуля упругости при растяжении и прочности при растяжении. Произведена оценка их точности на тренировочном и тестовом датасете, отобраны две наилучшие модели.
3. Сформированы 3 нейронные сети, дающие рекомендацию соотношения матрица-наполнитель. Проведена оценка точности прогноза на тренировочном и тестовом датасете, определена наилучшая нейросеть.
4. Разработано приложение с интерфейсом командной строки на основе двух лучших моделей машинного обучения, и одной нейронной сети, показавшей наилучшие результаты.

**Библиографический список**

1. Федеральный закон от 27.07.2006 N 149-ФЗ (ред. от 25.11.2017) «Об информации, информационных технологиях и о защите информации» (с изм. и доп., вступ. в силу с 01.01.2018)

2. Федеральный закон от 20.07.2004 N 72-ФЗ "О внесении изменений в Закон Российской Федерации "Об авторском праве и смежных правах" // Российская газета. — 2004. — № 159 (28 июля).

3. ГрасД. Data Science. Наука о данных с нуля: Пер. с англ. - 2-е изд., перераб. и доп. - СПб.: БХВ-Петербурr, 2021. - 416 с.: ил.

4. Силен Дэви, Мейсман Арно, Али Мохамед. Основы Data Science и Big Data. Python и наука о данных. – СПб.: Питер, 2017. – 336 с.: ил.

5. Композиционные материалы : учебное пособие для вузов / Д. А. Иванов, А. И. Ситников, С. Д. Шляпин ; под редакцией А. А. Ильина. — Москва : Издательство Юрайт, 2019 — 253 с. — (Высшее образование). — Текст : непосредственный.

6. Документация по языку программирования python: – Режим доступа: <https://docs.python.org/3.8/index.html>.

7. Документация по библиотеке matplotlib: – Режим доступа: <https://matplotlib.org/stable/users/index.html>.

8. Документация по библиотеке seaborn: – Режим доступа: <https://seaborn.pydata.org/tutorial.html>.

9. Документация по библиотеке sklearn: – Режим доступа: <https://scikit-learn.org/stable/user_guide.html>.

10. Документация по библиотеке keras: – Режим доступа: <https://keras.io/api/>.

11. Документация по библиотеке numpy: – Режим доступа: <https://numpy.org/doc/1.22/user/index.html#user>.

12. Документация по библиотеке pandas: – Режим доступа: <https://pandas.pydata.org/docs/user_guide/index.html#user-guide>.

13. Andre Ye. 5 алгоритмов регрессии в машинном обучении, о которых вам следует знать: – Режим доступа:<https://habr.com/ru/company/vk/blog/513842/>.

14. Alex Maszański. Метод k-ближайших соседей (k-nearest neighbour): – Режим доступа: <https://proglib.io/p/metod-k-blizhayshih-sosedey-k-nearest-neighbour-2021-07-19>.

15. Yury Kashnitsky. Открытый курс машинного обучения. Тема 3. Классификация, деревья решений и метод ближайших соседей: – Режим доступа: <https://habr.com/ru/company/ods/blog/322534/>.

16. Yury Kashnitsky. Открытый курс машинного обучения. Тема 5. Композиции: бэггинг, случайный лес: – Режим доступа: <https://habr.com/ru/company/ods/blog/324402/>.

17. Alex Maszański. Машинное обучение для начинающих: алгоритм случайного леса (Random Forest): – Режим доступа: <https://proglib.io/p/mashinnoe-obuchenie-dlya-nachinayushchih-algoritm-sluchaynogo-lesa-random-forest-2021-08-12>.

18. Alex Maszański. Решаем задачи машинного обучения с помощью алгоритма градиентного бустинга: – Режим доступа: <https://proglib.io/p/reshaem-zadachi-mashinnogo-obucheniya-s-pomoshchyu-algoritma-gradientnogo-bustinga-2021-11-25>.

19. Руководство по быстрому старту в flask: – Режим доступа: <https://flask-russian-docs.readthedocs.io/ru/latest/quickstart.html>.

20. Loginom Вики. Алгоритмы: – Режим доступа: <https://wiki.loginom.ru/algorithms.html>.

21. Friedman, Stochastic Gradient Boosting, 1999

22. T. Hastie, R. Tibshirani and J. Friedman. Elements of Statistical Learning Ed. 2, Springer, 2009

23. J. Friedman, Greedy Function Approximation: A Gradient Boosting Machine, The Annals of Statistics, Vol. 29, No. 5, 2001.