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 Сафин Ренат Халимович

Москва, 2023

# Содержание

Введение ........................................................................................................... 3

1. Аналитическая часть .................................................................................... 5

1.1. Постановка задачи ..................................................................................... 5

1.2. Описание используемых методов............................................................. 7

1.3. Разведочный анализ данных.................................................................... 22

2. Практическая часть .................................................................................... 29

2.1. Предобработка данных............................................................................ 29

2.2. Разработка и обучение моделей............................................................... 32

2.3. Результаты тестирования моделей.......................................................... 34

2.4. Тестирование модели............................................................................... 35

2.5. Разработка нейронной сети для рекомендации соотношения «матрица-наполнитель»............................................................................................................. 37

Заключение ..................................................................................................... 42

Библиографический список .......................................................................... 43

Приложение......................................................................................................45

# Введение

Тема данной работы - прогнозирование конечных свойств новых материалов (композиционных материалов).

Композиционными называют материалы, полученные из двух или более компонентов и состоящие из двух или более фаз. Один компонент (матрица) образует непрерывную фазу, другой является наполнителем. 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.

Матрица может быть металлической, керамической, углеродной. Наполнитель представлен в виде частиц и волокон, обладающих существенно более высокими физико-механическими свойствами, чем матрица. Частицы обычно называют дисперсным наполнителем, они имеют неопределенную, кубическую, шарообразную или чешуйчатую форму с размерами от долей мм до микронных и наноразмерных величин.

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.

Композиты имеют собственную классификацию:

1. По природе матрицы;
2. По природе и форме наполнителя;
3. По структуре полимерных композитов;
4. По степени ориентации наполнителя, анизотропии материала;
5. По методам изготовления материалов и изделий;
6. По количеству компонентов;
7. По объему содержания наполнителя;
8. По функциональности;

Возможность получить композиты с уникальными эксплуатационными свойствами способствует широкому распространению композиционных материалов в различных областях техники.

Учитывая такое широкое распространение и высокую потребность в новых материалах, тема данной работы является очень актуальной.

Расширение разнообразия материалов, используемых при проектировании нового композиционного материала, увеличивает необходимость определения свойств нового композита при минимальных финансовых затратах. Для решения этой проблемы обычно используются два способа: физические тесты образцов материалов или оценка свойств, в том числе на основе физико-математических моделей. Традиционно разработка композитных материалов является долгосрочным процессом, так как из свойств отдельных компонентов невозможно рассчитать конечные свойства композита. Для достижения определенных характеристик требуется большое количество различных комбинированных тестов, что делает насущной задачу прогнозирования успешного решения, снижающего затраты на разработку новых материалов и затраты на рабочую силу.

# 1. Аналитическая часть

## 1.1 Постановка задачи

В данной работе исследуется композит с матрицей из базальтопластика и нашивками из углепластика. От специалистов в предметной области был получен датасет, содержащий данные о свойствах матрицы и наполнителя, производственных параметрах и свойствах готового композита. От нас требуется разработать модели, прогнозирующие значения некоторых свойств в зависимости от остальных. Так же требуется разработать приложение, делающее удобным использование данных моделей специалистом предметной области.

Целевыми параметрами являются:

* Модуль упругости при растяжении,
* Прочность при растяжении,
* Соотношение матрица-наполнитель.

Датасет состоит из двух файлов: X\_bp (с данными о параметрах базальтопластика, состоящий из 1024 строки и 11 столбцов, включая индекс) и Х\_nup (данными нашивок углепластика, состоящий из 1041 строки и 4 столбцов, включая индекс).

Объединяем файлы с типом INNER по индексу. После объединения часть строк из файла X\_nup была отброшена. После объединения в ходе дальнейшего исследования выясняется, что часть строк из таблицы X\_nup не попала в датасет, осталось 1023 строки:

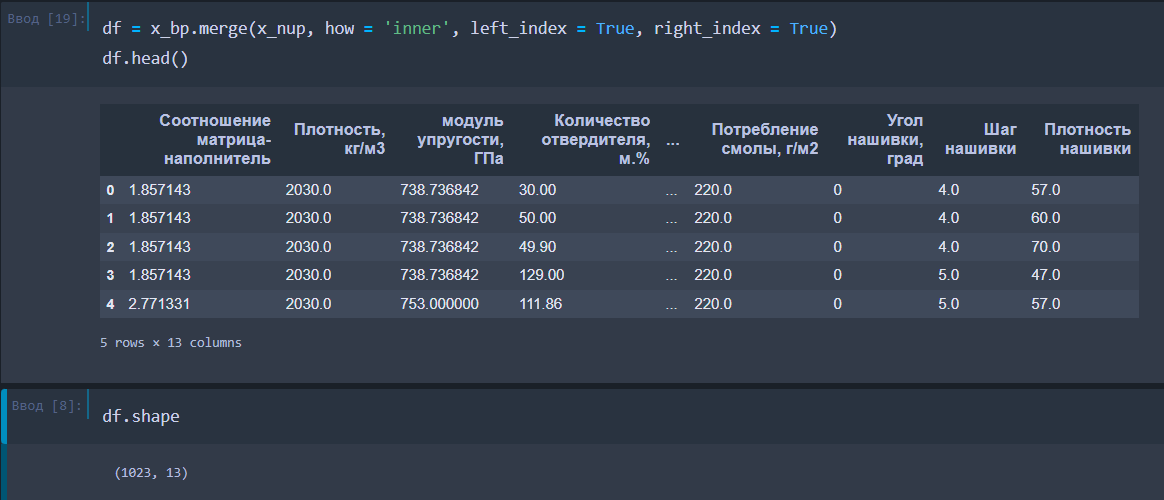


Рисунок 1 – Датасет после объединения X\_bp и Х\_nup

Все признаки имеют вещественный тип float64, исключение составляет одна колонка («Угол нашивки, град», int64 – целые).

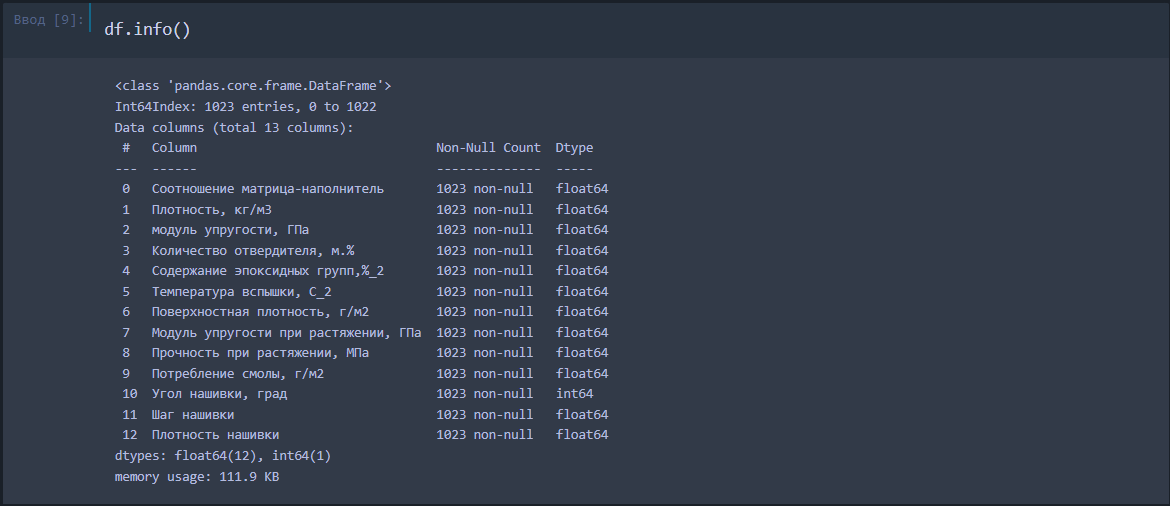


Рисунок 2 – Типы данных объединённого датасета

Пропусков в данных нет. Все признаки, кроме «Угол нашивки», являются непрерывными, количественными.

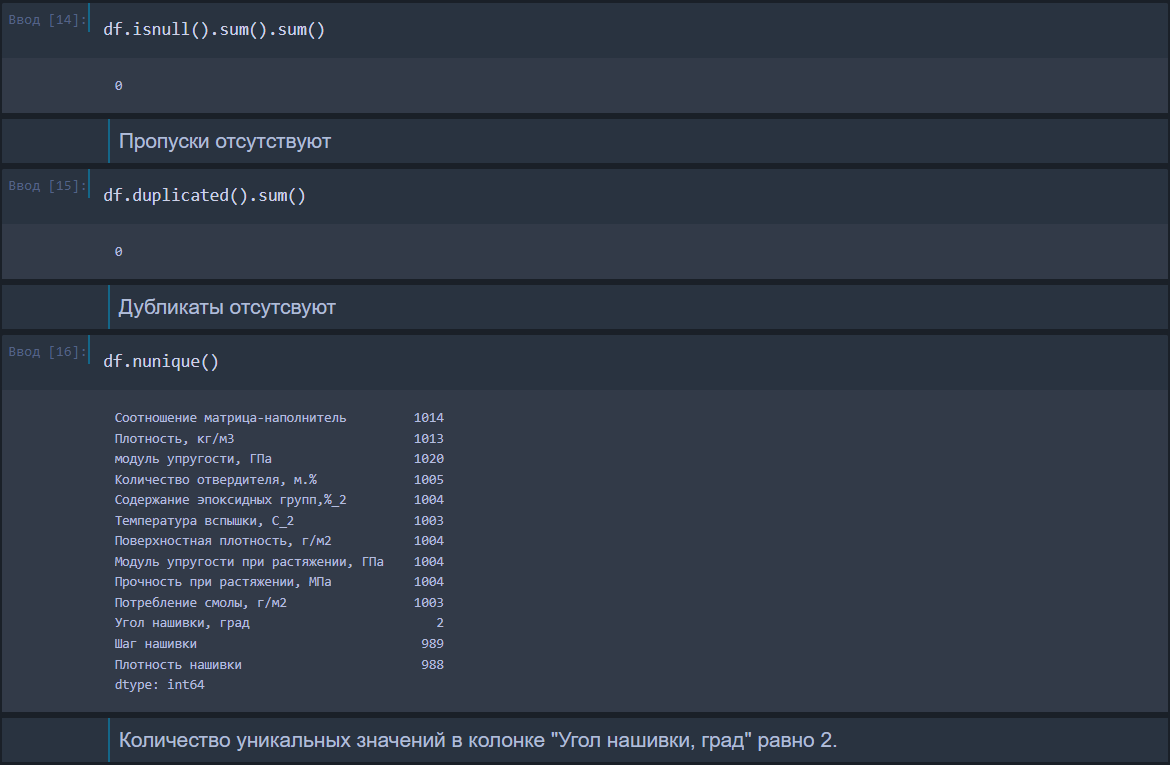


Рисунок 3 – Пропуски, дубликаты, уникальные значения датасета

Для каждой колонки получаем медианное значение, добавляем к описательной статистике:

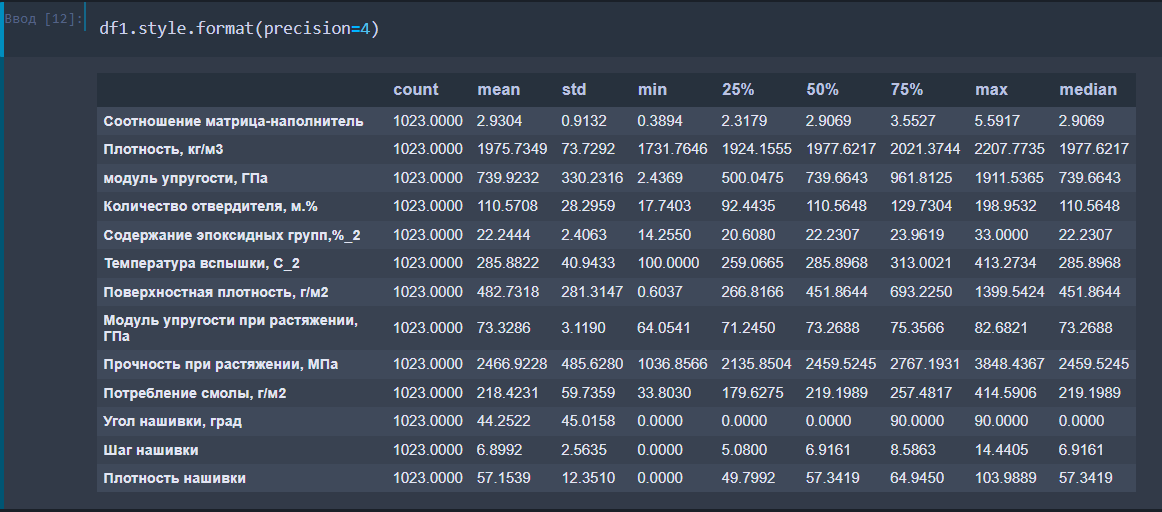


Рисунок 4 – Описательная статистика с добавлением медианного значения

При анализе отдельных признаков, таких как «Плотность нашивки», «Шаг нашивки» выяснилось, что среднее значение, медиана и большая часть значений из «топа» находятся очень близко друг другу. Причём значения либо не выходят вовсе, либо незначительно выходят за пределами 1 и 3 квартилей, и примерно настолько же отдалены от максимума. Возможно, если провести дальнейшее исследование на ненормализованных данных, можно будет найти другие закономерности, позволяющие выдвинуть гипотезы о происхождении и достоверности данных.

«Угол нашивки» принимает только два значения (0, 90). Используем LabelEncoder для кодировки, чтобы привести данные в этой колонке к значениям 0 и 1, превратив в категориальные(бинарные) признаки.

## 1.2 Описание используемых методов

Данная задача в рамках классификации категорий машинного обучения относится к машинному обучению с учителем и традиционно это задача регрессии. Цель любого алгоритма обучения с учителем — определить функцию потерь и минимизировать её, поэтому для наилучшего решения в процессе исследования были применены следующие методы:

* линейная регрессия;
* К-ближайших соседей;
* стохастический градиентный спуск;
* метод опорных векторов;
* случайный лес;
* градиентный бустинг;
* многослойный перцептрон.

Линейная регрессия (Linear regression) — это алгоритм машинного обучения, основанный на контролируемом обучении, рассматривающий зависимость между одной входной и выходными переменными. Это один из самых простых и эффективных инструментов статистического моделирования. Она определяет зависимость переменных с помощью линии наилучшего соответствия. Модель регрессии создаёт несколько метрик. R2 , или коэффициент детерминации, позволяет измерить, насколько модель может объяснить дисперсию данных. Если R-квадрат равен 1, это значит, что модель описывает все данные. Если же R-квадрат равен 0,5, модель объясняет лишь 50 процентов дисперсии данных. Оставшиеся отклонения не имеют объяснения. Чем ближе R2 к единице, тем лучше.

Достоинства метода: быстр и прост в реализации; легко интерпретируем; имеет меньшую сложность по сравнению с другими алгоритмами;

Недостатки метода: моделирует только прямые линейные зависимости; требует прямую связь между зависимыми и независимыми переменными; выбросы оказывают огромное влияние, а границы линейны.



Рисунок 5 – График линейной регрессии для модуля упругости и для прочности при растяжении

Метод ближайших соседей - К-ближайших соседей (kNN - k Nearest Neighbours) ищет ближайшие объекты с известными значения целевой переменной и основывается на хранении данных в памяти для сравнения с новыми элементами. Алгоритм находит расстояния между запросом и всеми примерами в данных, выбирая определенное количество примеров (k), наиболее близких к запросу, затем голосует за наиболее часто встречающуюся метку (в случае задачи классификации) или усредняет метки (в случае задачи регрессии).

Достоинства метода: прост в реализации и понимании полученных результатов; имеет низкую чувствительность к выбросам; не требует построения модели; допускает настройку нескольких параметров; позволяет делать дополнительные допущения; универсален; находит лучшее решение из возможных; решает задачи небольшой размерности.

Недостатки метода: замедляется с ростом объёма данных; не создаёт правил; не обобщает предыдущий опыт; основывается на всем массиве доступных исторических данных; невозможно сказать, на каком основании строятся ответы; сложно выбрать близость метрики; имеет высокую зависимость результатов классификации от выбранной метрики; полностью перебирает всю обучающую выборку при распознавании; имеет вычислительную трудоёмкость.



Рисунок 6 – График К-ближайших соседей для модуля упругости и для прочности при растяжении

Стохастический градиентный спуск (SGDRegressor) — это простой, но очень эффективный подход к подгонке линейных классификаторов и регрессоров под выпуклые функции потерь. Этот подход подразумевает корректировку весов нейронной сети, используя аппроксимацию градиента функционала, вычисленную только на одном случайном обучающем примере из выборки.

Достоинства метода: эффективен; прост в реализации; имеет множество возможностей для настройки кода; способен обучаться на избыточно больших выборках.

Недостатки метода: требует ряд гиперпараметров; чувствителен к масштабированию функций; может не сходиться или сходиться слишком медленно; функционал многоэкстремален; процесс может "застрять" в одном из локальных минимумов; возможно переобучение.



Рисунок 7 – График стохастического градиентного спуска для модуля упругости и для прочности при растяжении

Метод опорных векторов (Support Vector Regression) – этот бинарный линейный классификатор был выбран, потому что он хорошо работает на небольших датасетах. Данный алгоритм – это алгоритм обучения с учителем, использующихся для задач классификации и регрессионного анализа, это контролируемое обучение моделей с использование схожих алгоритмов для анализа данных и распознавания шаблонов. Учитывая обучающую выборку, где алгоритм помечает каждый объект, как принадлежащий к одной из двух категорий, строит модель, которая определяет новые наблюдения в одну из категорий.

Модель метода опорных векторов – отображение данных точками в пространстве, так что между наблюдениями отдельных категорий имеется разрыв, и он максимален.

Каждый объект данных представляется как вектор (точка) в p-мерном пространстве. Он создаёт линию или гиперплоскость, которая разделяет данные на классы.

Достоинства метода: для классификации достаточно небольшого набора данных. При правильной работе модели, построенной на тестовом множестве, вполне возможно применение данного метода на реальных данных. Эффективен при большом количестве гиперпараметров. Способен обрабатывать случаи, когда гиперпараметров больше, чем количество наблюдений. Существует возможность гибко настраивать разделяющую функцию.  Алгоритм максимизирует разделяющую полосу, которая, как подушка безопасности, позволяет уменьшить количество ошибок классификации.

Недостатки метода: неустойчивость к шуму, поэтому в работе была проведена тщательнейшая работа с выбросами, иначе в обучающих данных шумы становятся опорными объектами-нарушителями и напрямую влияют на построение разделяющей гиперплоскости; для больших наборов данных требуется долгое время обучения; достаточно сложно подбирать полезные преобразования данных; параметры модели сложно интерпретировать, поэтому были рассмотрены и другие методы.



Рисунок 8 – График опорных векторов для модуля упругости и для прочности при растяжении

Случайный лес (RandomForest) — это множество решающих деревьев. Универсальный алгоритм машинного обучения с учителем, представитель ансамблевых методов.  Если точность дерева решений оказалось недостаточной, мы можем множество моделей собрать в коллектив.

Достоинства метода: не переобучается; не требует предобработки входных данных; эффективно обрабатывает пропущенные данные, данные с большим числом классов и признаков; имеет высокую точность предсказания и внутреннюю оценку обобщающей способности модели, а также высокую параллелизуемость и масштабируемость.

Недостатки метода: построение занимает много времени; сложно интерпретируемый; не обладает возможностью экстраполяции; может недо обучаться; трудоёмко прогнозируемый; иногда работает хуже, чем линейные методы.

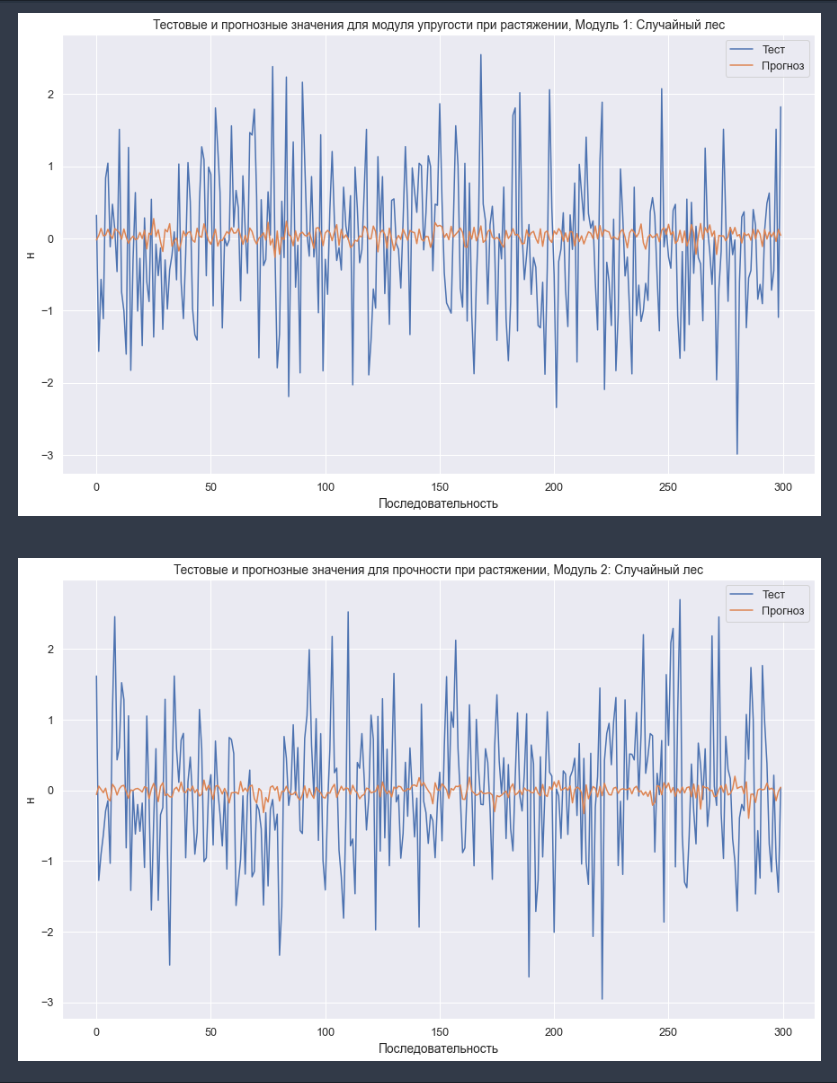


Рисунок 9 – График «случайного леса» для модуля упругости и для прочности при растяжении

Градиентный бустинг (Gradient Boosting) — это ансамбль деревьев решений, обученный с использованием градиентного бустинга. В основе данного алгоритма лежит итеративное обучение деревьев решений с целью минимизировать функцию потерь. Основная идея градиентного бустинга: строятся последовательно несколько базовых классификаторов, каждый из которых как можно лучше компенсирует недостатки предыдущих. Финальный классификатор является линейной композицией этих базовых классификаторов.

Достоинства метода: новые алгоритмы учатся на ошибках предыдущих; требуется меньше итераций, чтобы приблизиться к фактическим прогнозам; наблюдения выбираются на основе ошибки; прост в настройке темпа обучения и применения; легко интерпретируем.

Недостатки метода: необходимо тщательно выбирать критерии остановки, иначе это может привести к переобучению; наблюдения с наибольшей ошибкой появляются чаще; слабее и менее гибок чем нейронные сети.

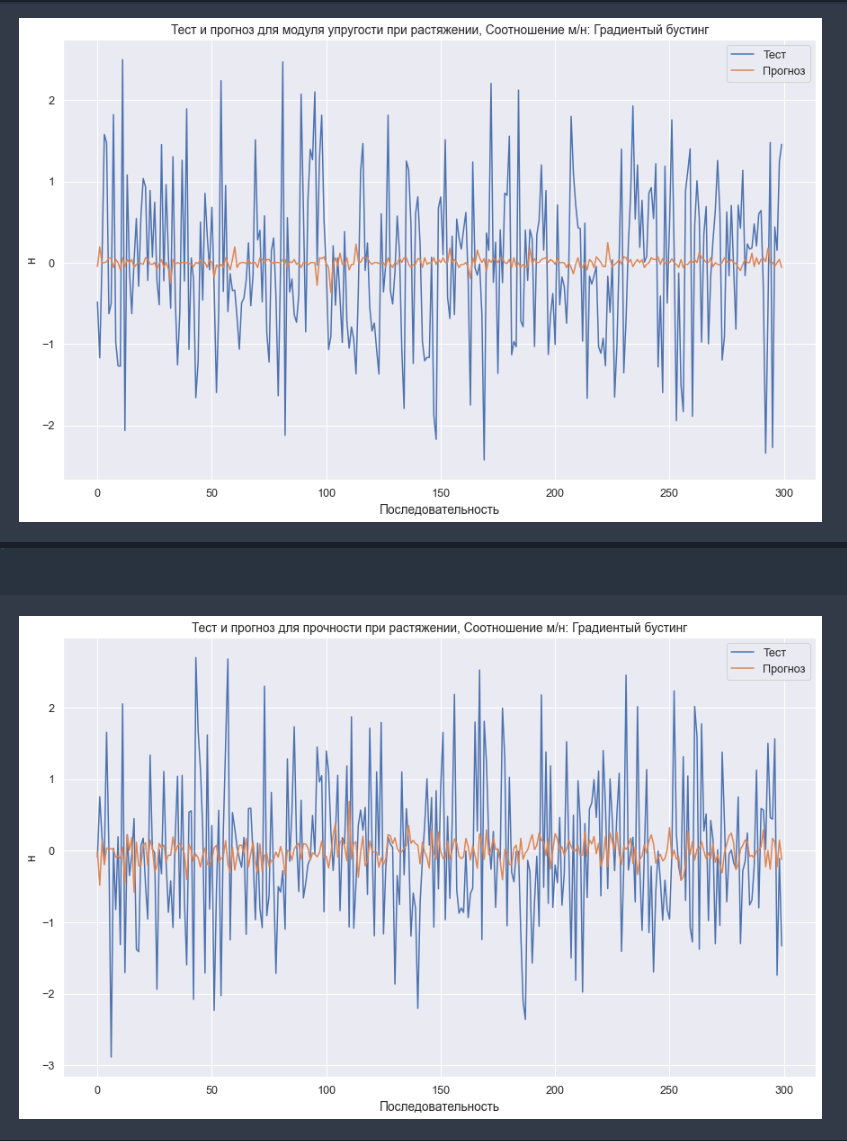


Рисунок 10 – График градиентного бустинга для модуля упругости и для прочности при растяжении

Многослойный персептрон (MLPRegressor) — это алгоритм обучения с учителем, который изучает функцию f(⋅):Rm→Ro обучением на наборе данных, где m — количество измерений для ввода и o- количество размеров для вывода. Это искусственная нейронная сеть, имеющая 3 или более слоёв персептронов. Эти слои - один входной слой, 1 или более скрытых слоёв и один выходной слой персептронов.

Достоинства метода: построение сложных разделяющих поверхностей; возможность осуществления любого отображения входных векторов в выходные; легко обобщает входные данные; не требует распределения входных векторов; изучает нелинейные модели.

Недостатки метода: имеет невыпуклую функцию потерь; разные инициализации случайных весов могут привести к разной точности проверки; требует настройки ряда гиперпараметров; чувствителен к масштабированию функций.



Рисунок 11 – График многослойного персептрона для модуля упругости и для прочности при растяжении

Качество моделей будем оценивать по следующим метрикам.

MSE (Mean Squared Error) или средняя квадратичная ошибка принимает значениях в тех же единицах, что и целевая переменная. Чем ближе к нулю MSE, тем лучше работают предсказательные качества модели.

MAE (Mean Absolute Error) или средняя абсолютная ошибка измеряет среднюю сумму абсолютной разницы между фактическим значением и прогнозируемым значением. Чем ниже MAE, тем лучше модель соответствует набору данных.

R2 или коэффициент детерминации измеряет долю дисперсии, объяснённую моделью, в общей дисперсии целевой переменной. Если он близок к единице, то модель хорошо объясняет данные, если же он близок к нулю, то качество прогноза идентично средней величине целевой переменной (т.е. очень низкое). Отрицательные значение коэффициента детерминации означают плохую объясняющую способность модели.

## 1.3 Разведочный анализ предложенных данных

Гистограммы распределения переменных и диаграммы «ящик с усами» приведены на рисунках 12-14. Все признаки, кроме «Угол нашивки», имеют нормальное распределение и принимают неотрицательные значения.

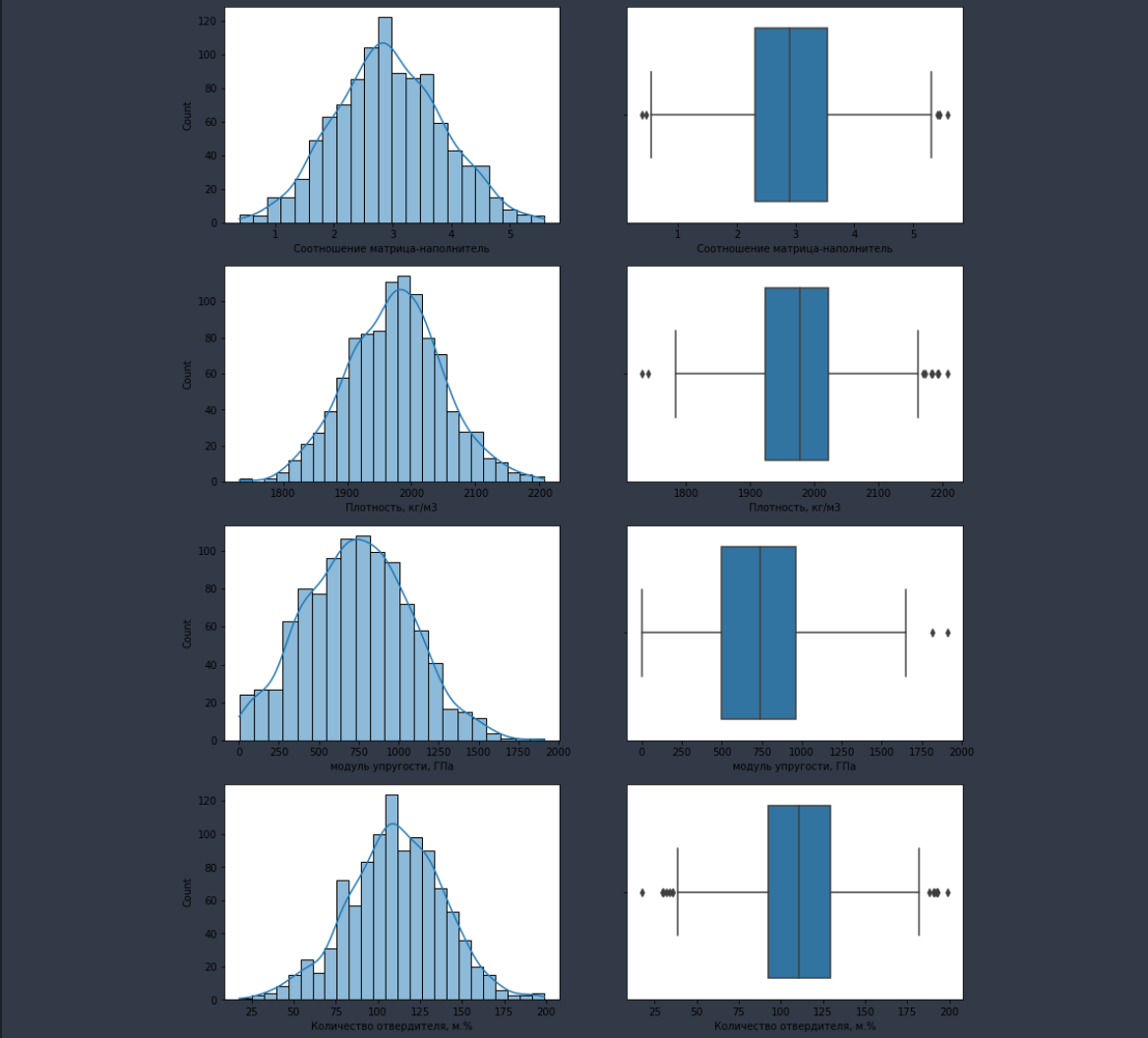


Рисунок 12 - Гистограммы распределения переменных

и диаграммы «ящик с усами» ч. 1

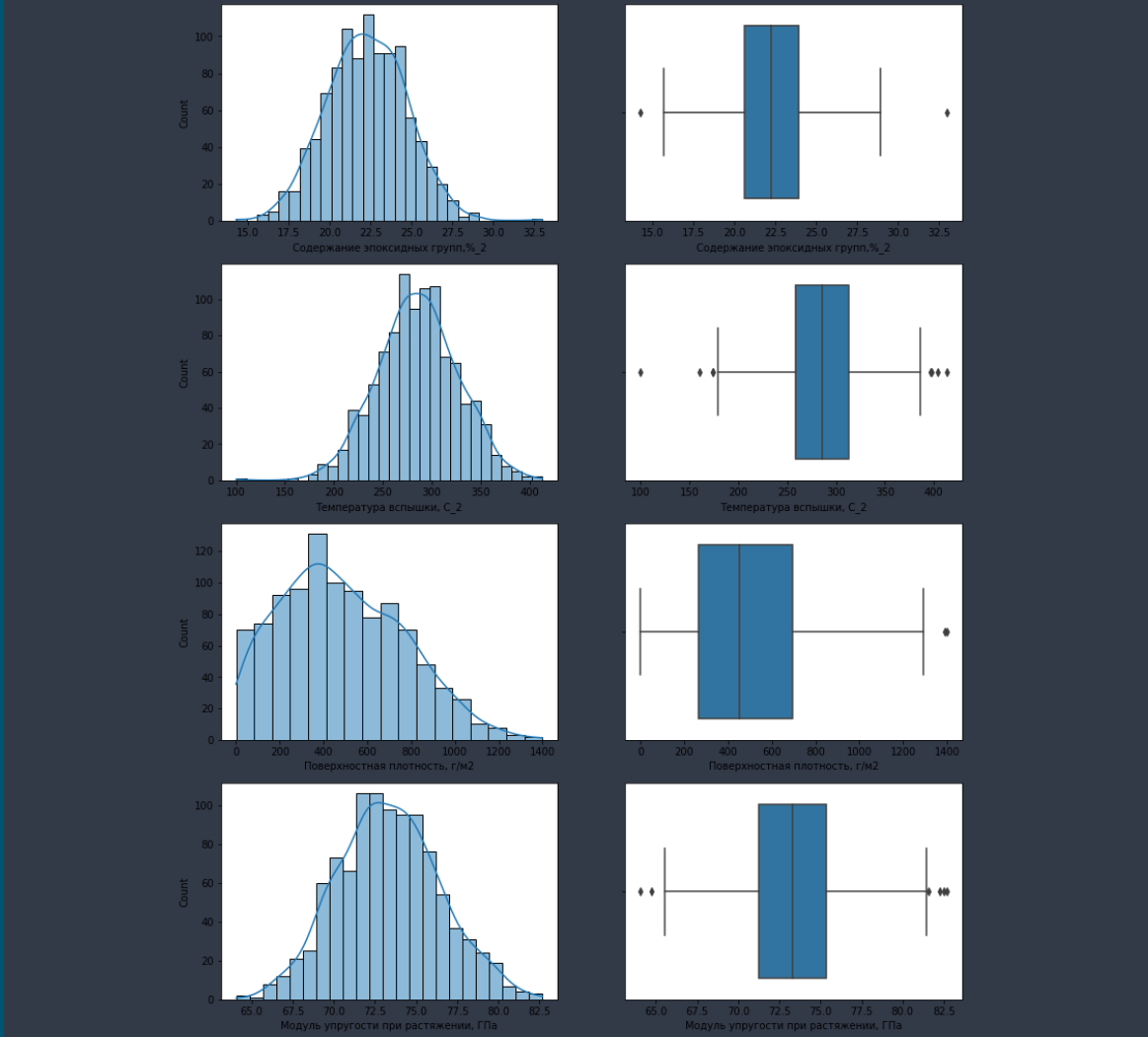


Рисунок 13 - Гистограммы распределения переменных

и диаграммы «ящик с усами» ч. 2

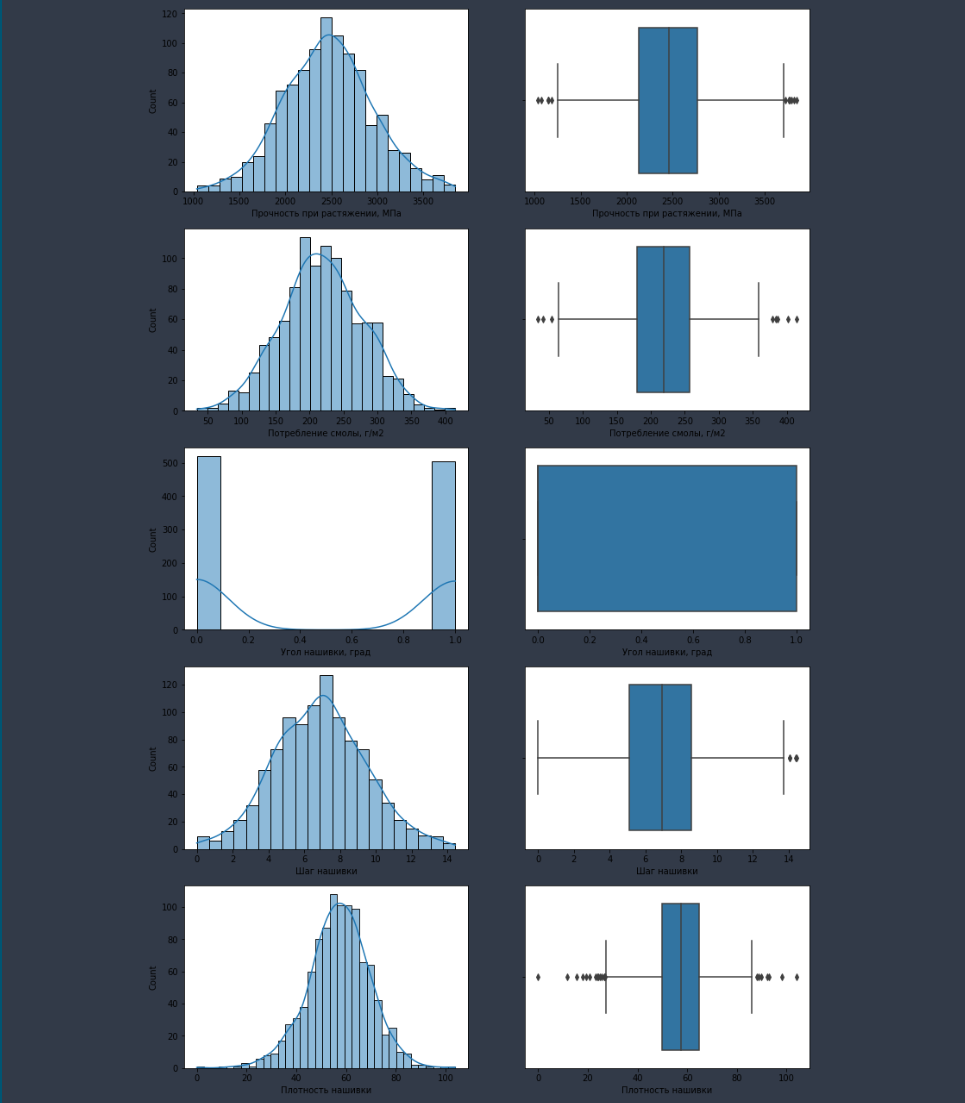


Рисунок 14 - Гистограммы распределения переменных

и диаграммы «ящик с усами» ч. 3

Исходя из того, что в датасете отсутствуют пропуски, дубликаты, значения некорректного типа, можем сделать предположение о том, что датасет уже был ранее подготовлен.

Попарные графики рассеяния точек приведены на рисунке 4. В качестве столбца для кодирования цвета берём(hue) берём «Угол нашивки, град»:

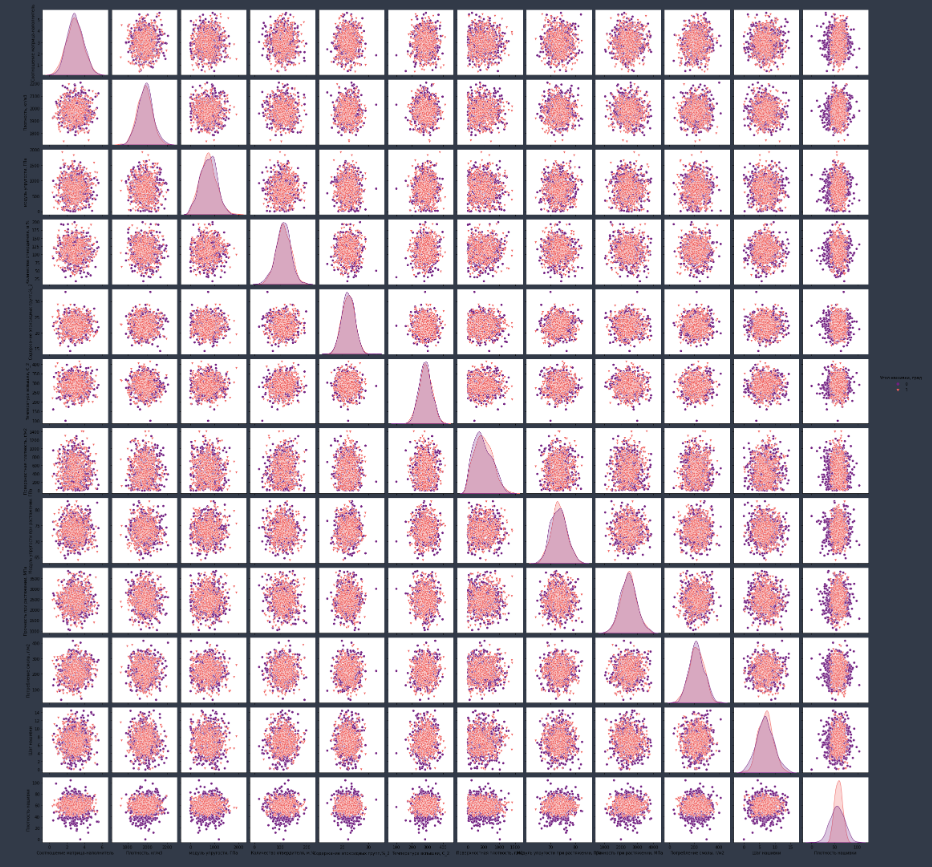


Рисунок 15 — Попарные графики рассеяния точек

По графикам рассеяния мы видим, что некоторые точки отстоят далеко от общего облака. Делаем вывод, что на графике присутствуют выбросы.

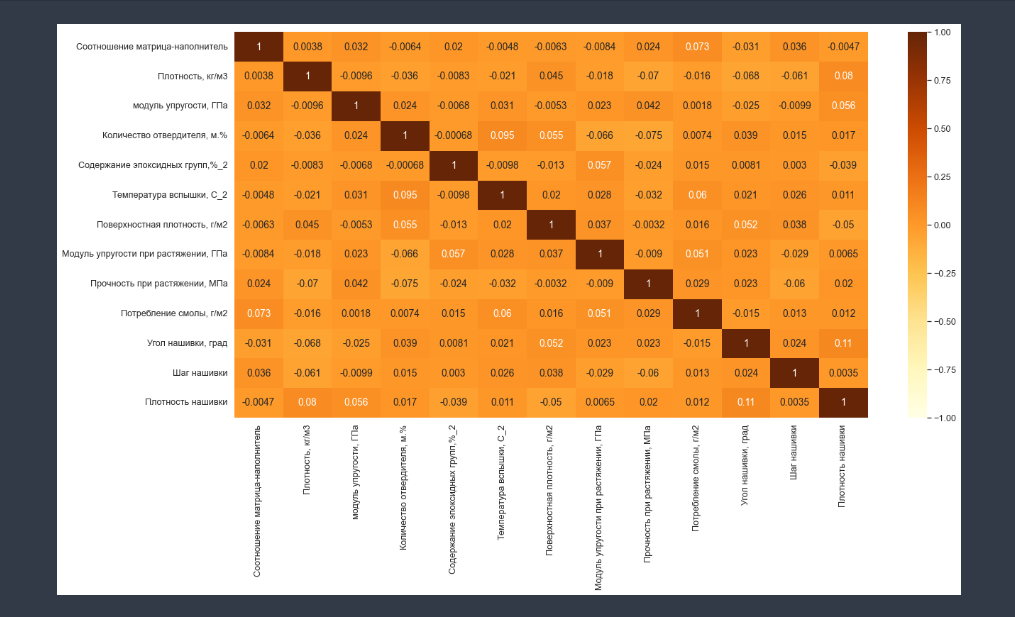


Рисунок 16 — Матрица корреляции

Как видно из матрицы корреляции, а также из диаграмм рассеяния зависимость между признаками практически не прослеживается.

# 2. Практическая часть

## 2.1 Предобработка данных

Необработанные данные могут содержать искажения и пропущенные значения, что способно привести к крайне неверным результатам по итогам моделирования. Для определения и удаления выбросов используем метод z-оценки и метод межквартильного диапазона.

Метод Z-оценки (3-х сигм). Результат можно увидеть на рисунке 17, к выбросам было отнесено 24 значения.

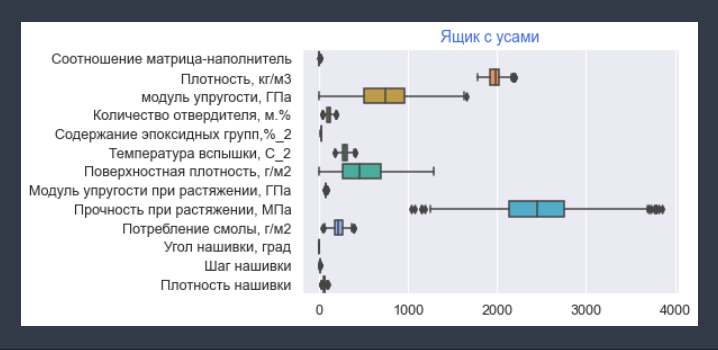


Рисунок 17 — Ящик с усами после удаления выбросов методом Z-оценки

Метод межквартильного диапазона. К выбросам было отнесено 87 значений, на рисунке 18 показан результат.

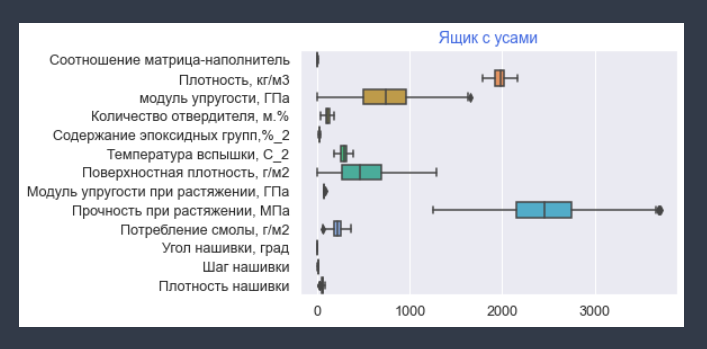


Рисунок 18 — Ящик с усами после удаления выбросов методом

межквартильного диапазона

Ранее было выдвинуто предположение, что датасет уже является подготовленным, поэтому логично предположить, что количество выбросов в нём минимально. Поэтому принимаем решение об очистке датасета по методу 3-х сигм. После этого в датасете остаётся 999 строк и 13 колонок – признаков. В задании целевыми переменными указаны: модуль упругости при растяжении, Гпа; прочность при растяжении, МПа; соотношение матрица-наполнитель.

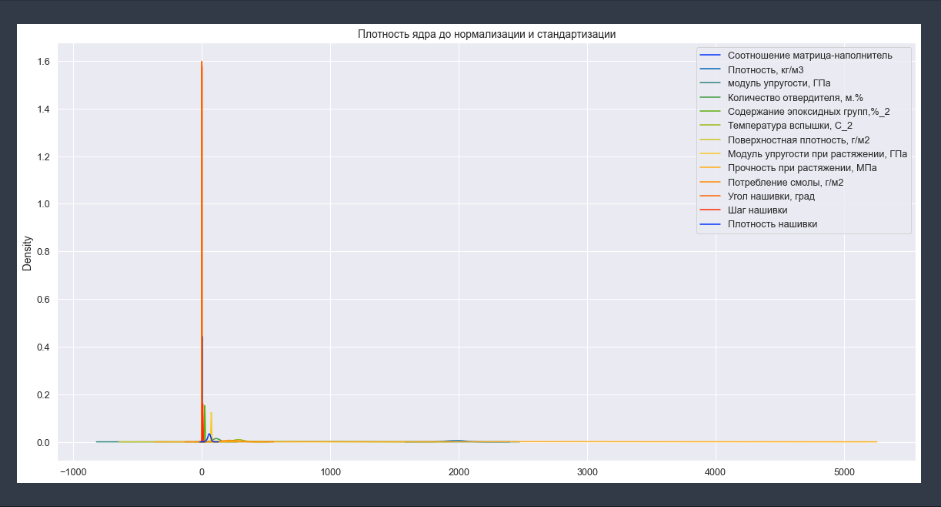


Рисунок 19 — Визуализация данных до нормализации и стандартизации

Проблема вещественных признаков в том, что их значения лежат в разных диапазонах, в разных масштабах. Для успешного обучения моделей необходимо провести нормализацию датасета. Используем MinMaxScaler()

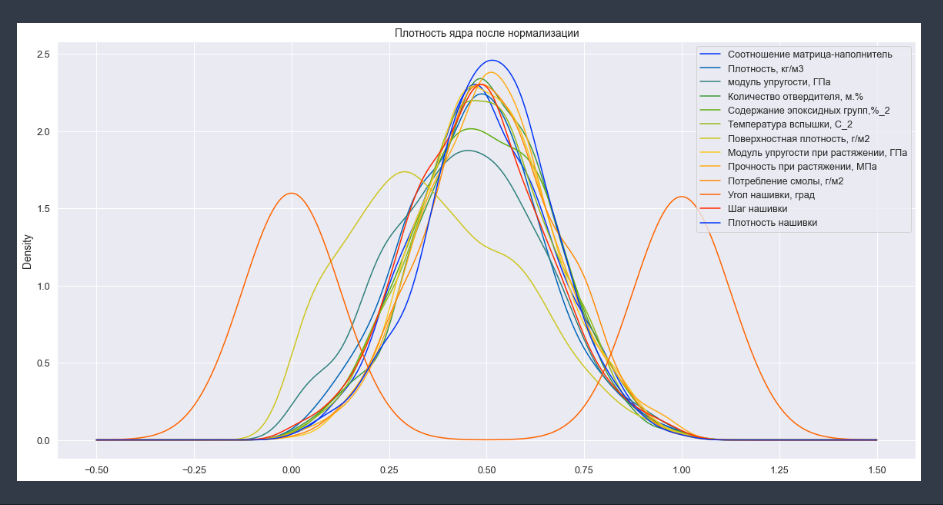


Рисунок 20 — Визуализация данных после нормализации

Для моделей Машинного обучения, основанных на вычислении дистанции между точками на графике как основу прогнозирования: Метод k-ближайших соседей (KNN), Метод опорных векторов (SVM), Дерево решений (Decision Tree) и др. будем использовать датасет после стандартизации. Такое преобразование необходимо, поскольку признаки датасета могут иметь большие различия между своими диапазонами. Используем StandardScaler()

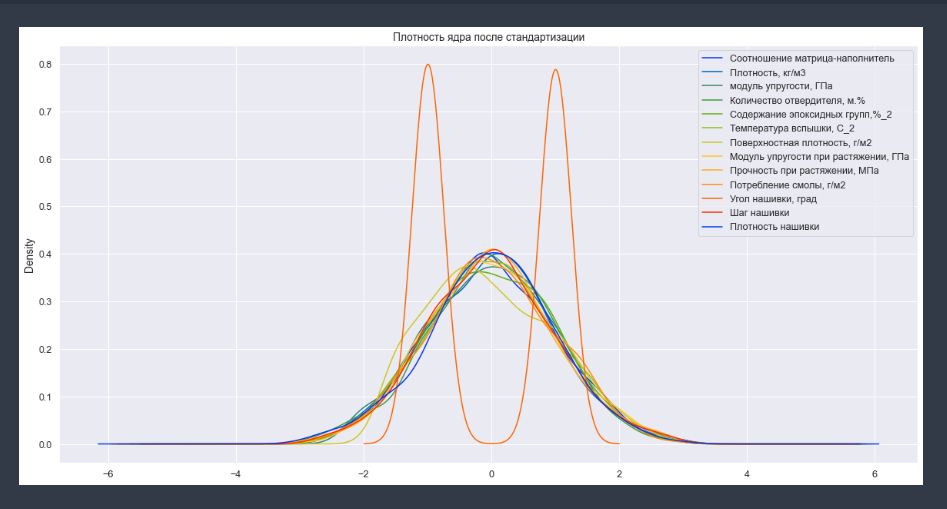


Рисунок 21 — Визуализация данных после стандартизации

## 2.2 Разработка и обучение моделей

Разработка и обучение моделей машинного обучения осуществлялась для двух выходных параметров: «Прочность при растяжении» и «Модуль упругости при растяжении» отдельно. Для решения применяются все методы, описанные выше.

При построении модели 30% данных выделяется на тестирование модели, остальная часть используется для обучения моделей. При построении моделей будет проведён поиск гиперпараметров модели с помощью поиска по сетке с перекрестной проверкой, количество блоков равно 10. После определения осуществляется подстановка оптимальных гиперпараметров в модель и обучение модели на тренировочных данных.

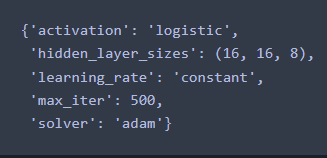


Рисунок 22 — Подобранные с помощью GridSearchCV гиперпараметры

В качестве параметра оценки выбраны: средняя квадратичная ошибка, средняя абсолютная ошибка, коэффициент детерминации. Далее проводится оценка полученных данных и сравнение со стандартными значениями.

Создаём список для хранения лучших моделей и датасет для хранения ошибок, отдельно для модуля упругости и для прочности при растяжении. Добавляем базовую модель на основе среднего значения (mean) для сравнения. Отобразим финальный результат с обученными моделями регрессии.

## 2.3 Результаты тестирования моделей



Рисунок 23 — Датасет для хранения ошибок для модуля упругости и прочности при растяжении

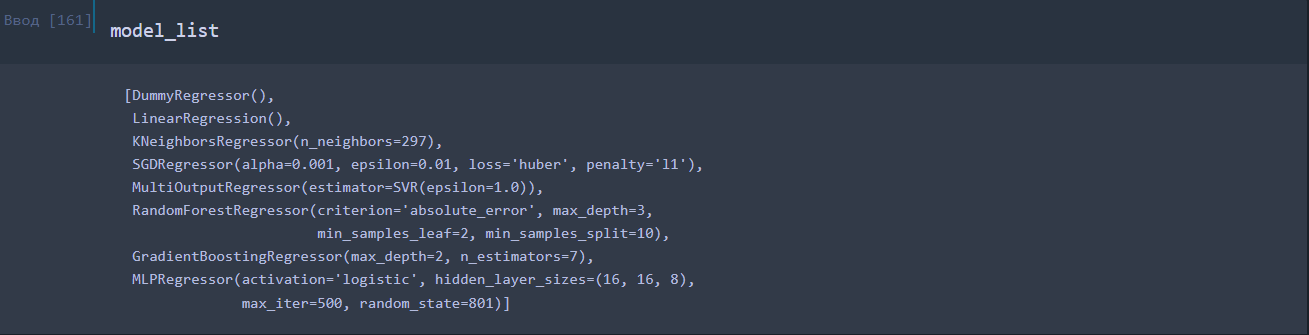


Рисунок 24 —Список моделей для модуля упругости и прочности при растяжении

Модели после настройки гиперпараметров показали результаты немного лучше. Однако, всеравно уступают базовой модели. Прочность при растяжении и модуль упругости не имеет линейной зависимости. Все использованные модели не справились с задачей. Свойства композитных материалов в первую очередь зависят от используемых материалов.

Лишь многослойный перцептрон дал результат лучший, чем базовая модель (на базе усреднения).

## 2.4 Тестирование модели

Снова разбиваем объединённый датасет с удалёнными выбросами на тестовую, тренировочную выборки, выделяя целевую переменную, отдельно для модуля упругости, отдельно для прочности. В этот раз используем Normalizer().

Поскольку наиболее удачной моделью был признан MLPregressor, то в качестве эксперимента и для поиска наиболее эффективной конфигурации модели создадим и обучим 2 новые модели на новых выборках.

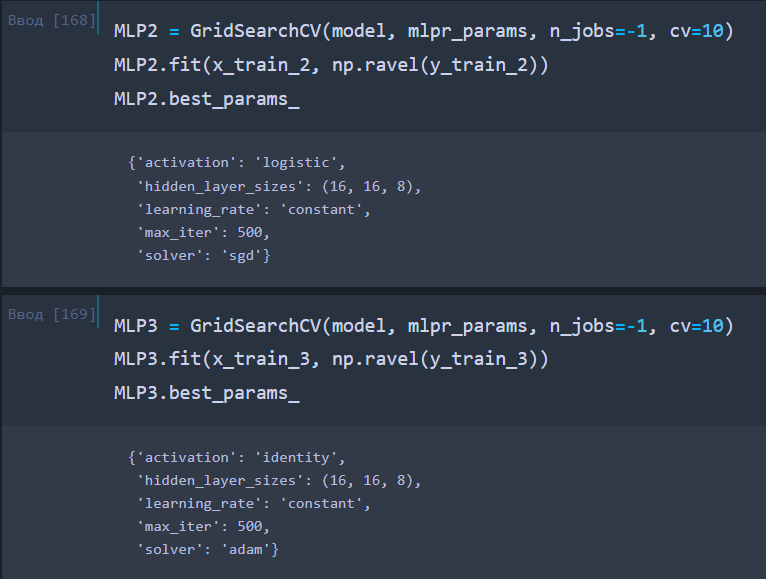


Рисунок 25 — Две модели с подобранными по GridSearchCV() параметрами

Протестируем все четыре модели на каждой из выборок

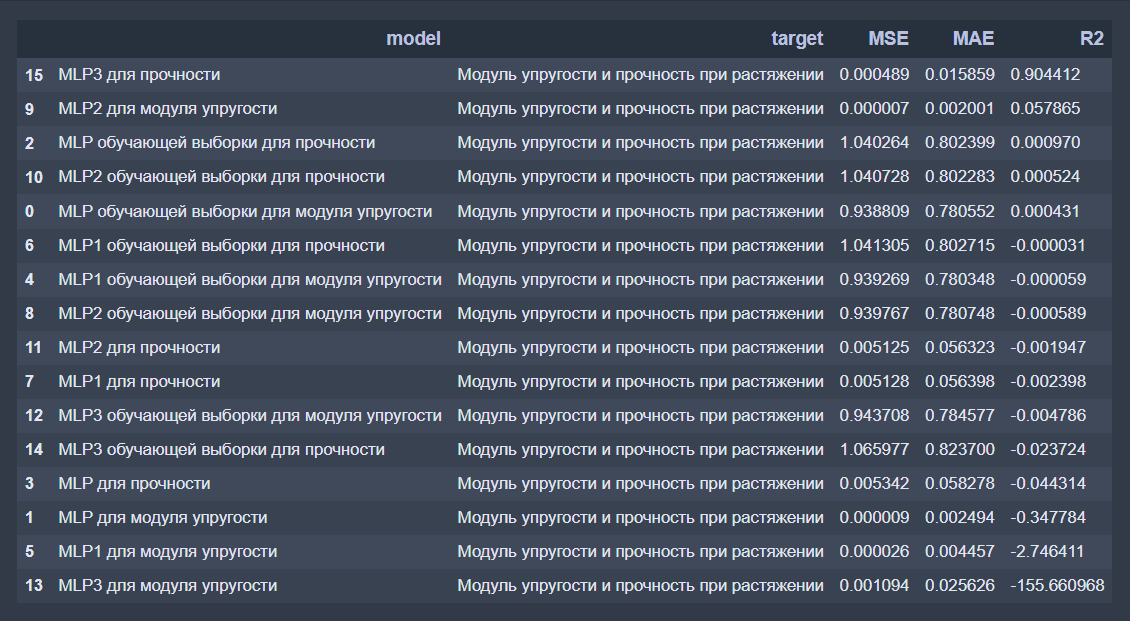


Рисунок 26 — Результаты проверки моделей

Самой нестабильной моделью оказывается MLP3,она прекрасно показывает себя при прогнозировании «прочности» для выборки на которой модель обучалась, при это не показывает каких-либо успехов на других выборках. На первый взгляд MLP2 показывает наилучший результат на фоне остальных.

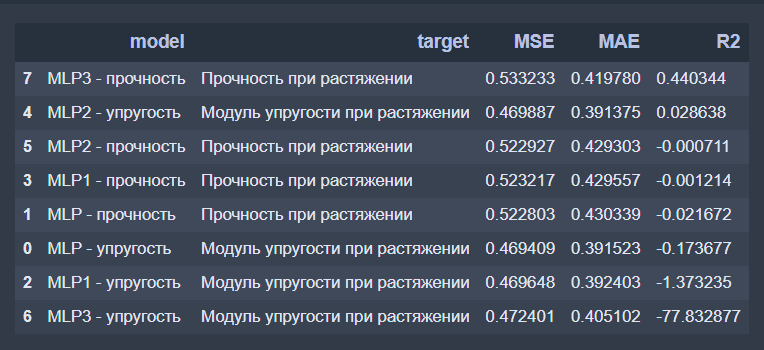


Рисунок 27 — Результаты проверки моделей: среднее для прочности и упругости моделей

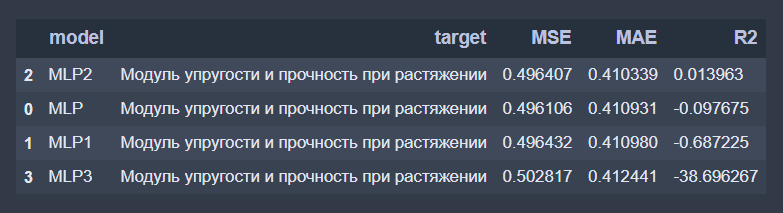


Рисунок 28 — Результаты проверки моделей: среднее для каждой из моделей

И действительно модель MLP2 является самой стабильной, показывая наиболее точные результаты, поэтому именно эту модели признаём наиболее удачной. При этом модели, обучавшиеся для прогнозирования целевой переменной – упругости показывают лучшие результаты, чем модели, обучавшиеся на целевой – прочности. Возможно, зависимость между упругостью и другими признаками выше, чем прочность.

## 2.5 Разработка нейронной сети для рекомендации соотношения «матрица-наполнитель»

Обучение нейронной сети — это такой процесс, при котором происходит подбор оптимальных параметров модели, с точки зрения минимизации функцио-нала ошибки. Начнём стоить нейронную сеть с помощью класса keras.Sequential.

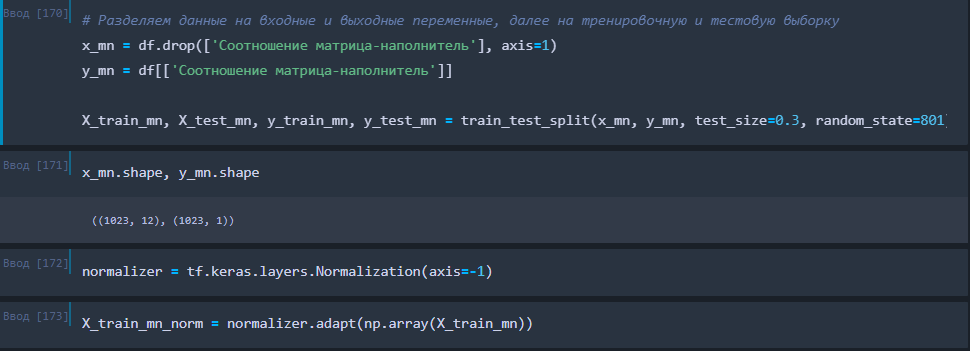


Рисунок 29 — Создание нейронной сети

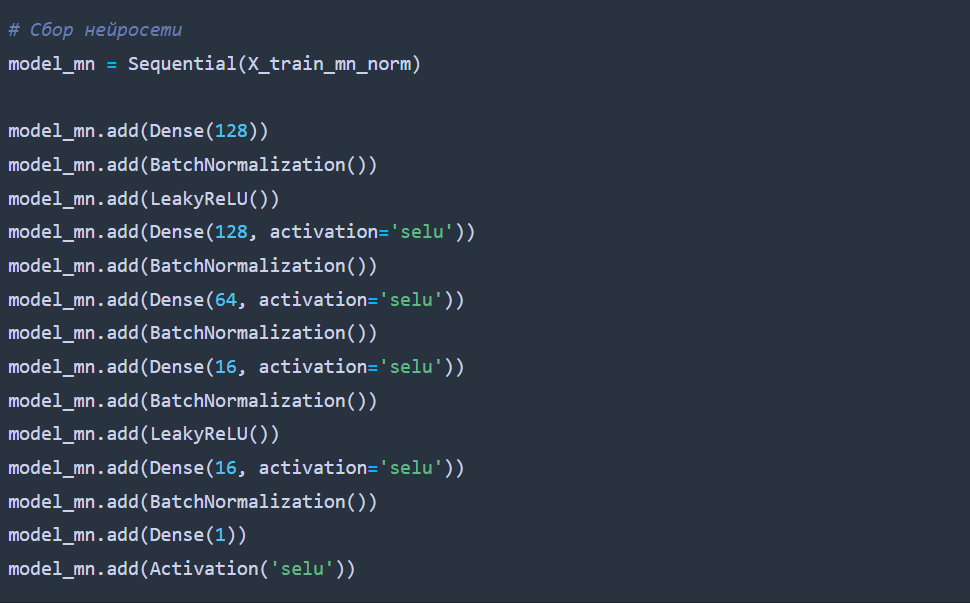


Рисунок 30 — Сбор нейросети

Определим параметры, посмотрим на результаты. Компилируем и обучаем модель.

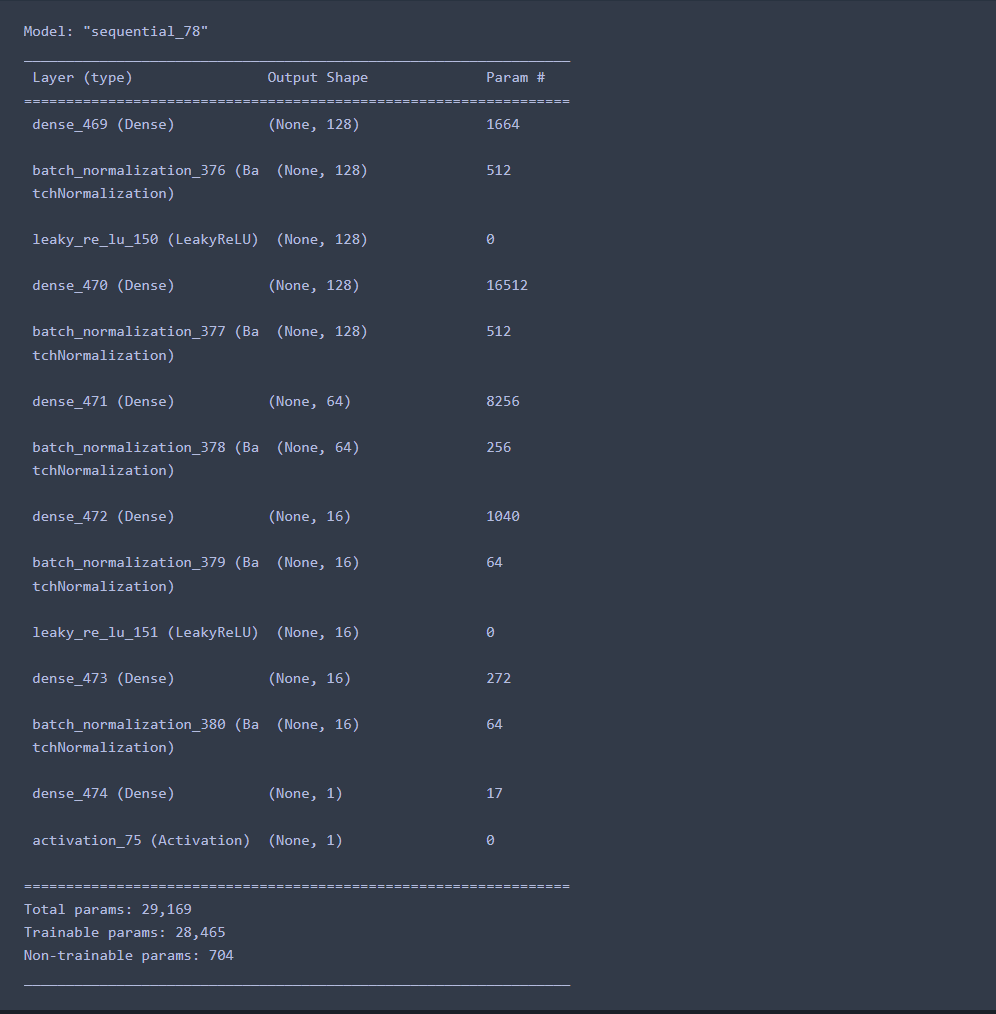


Рисунок 31 — Архитектура нейронной сети

Оценим модель, посмотрим на потери, зададим функцию для визуализации факт/прогноз для результатов моделей.

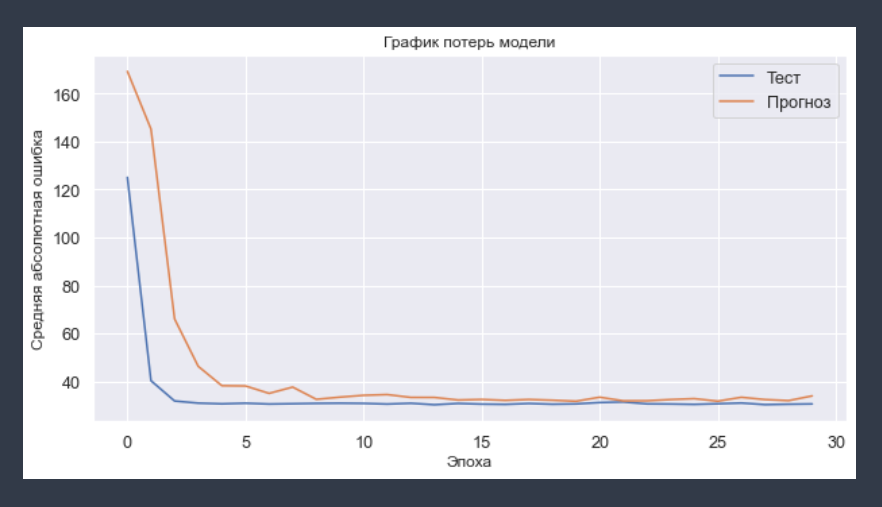


Рисунок 32 — График потерь модели на тренировочной и тестовой выборках

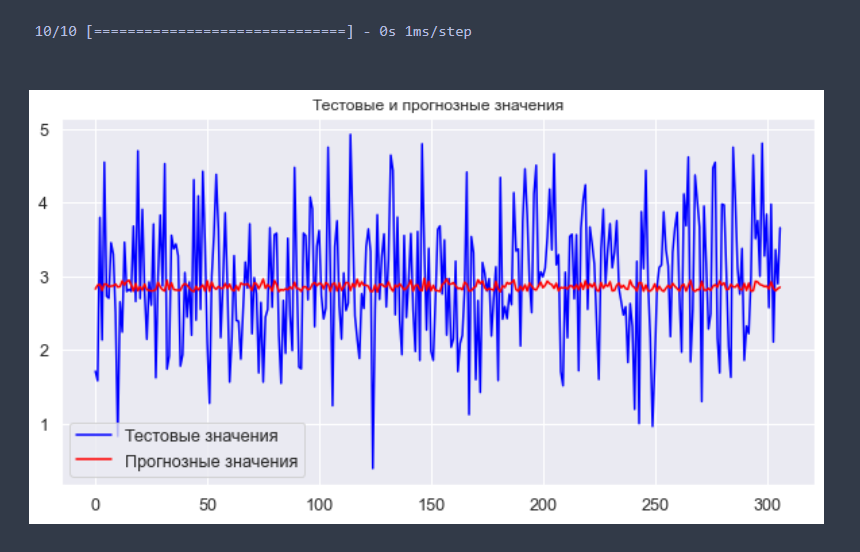


Рисунок 33 — График оригинального и предсказанного значения



Рисунок 34 — График рассеяния оригинального и предсказанного значения

Проведём тестирование нейросети:

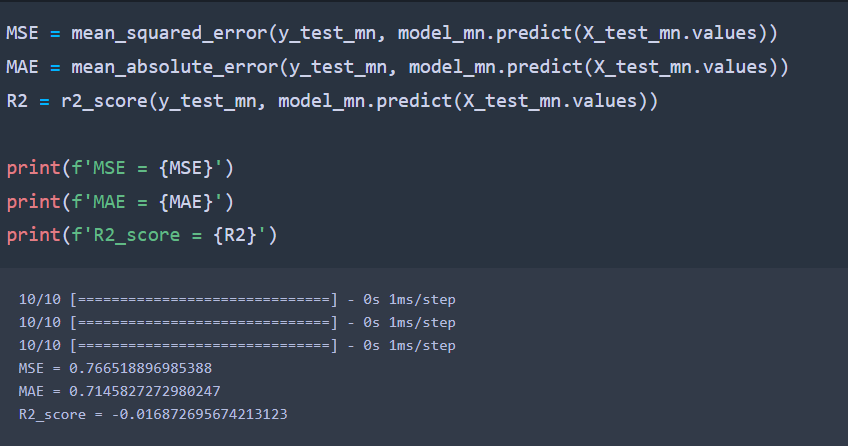


Рисунок 35 — Тестирование модели

# Заключение

Данная исследовательская работа позволяет сделать некоторые основные выводы по теме. Распределение полученных данных в объединённом датасете близко к нормальному, но коэффициенты корреляции между парами признаков стремятся к нулю. Использованные при разработке моделей подходы не позволили получить сколько-нибудь достоверных прогнозов. Применённые модели регрессии не показали высокой эффективности в прогнозировании свойств композитов. Лучшие метрики показал многослойный персептрон (MLPregressor).

Был сделан вывод, что невозможно определить из свойств материалов соотношение «матрица – наполнитель». Данный факт не указывает на то, что прогнозирование характеристик композитных материалов на основании предоставленного набора данных невозможно, но может указывать на недостатки базы данных, подходов, использованных при прогнозе, необходимости пересмотра инструментов для прогнозирования.

Необходимы дополнительные вводные данные, получение новых результирующих признаков в результате математических преобразований, консультации экспертов предметной области, новые исследования. Для решения данной задачи требуется подключить целую команду специалистов.

В целом прогнозирование конечных свойств композитных материалов без изучения материаловедения, погружения в вопрос экспериментального анализа характеристик композитных материалов не демонстрирует сколько-нибудь удовлетворительных результатов. Проработка моделей и построение прогнозов требует внедрения в процесс производных от имеющихся показателей для выявления иного уровня взаимосвязей. Отсюда, также учитывая отсутствие корреляции между признаками, делаем вывод, что текущим набором алгоритмов задача либо не решается, либо решение будет найдено при добавлении новых, возможно, ещё неизвестных переменных.

# Библиографический список

1 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.

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

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

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

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

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

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

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>.

# Приложение

Для данного исследования был создан удаленный репозиторий на GitHub, который находится по адресу <https://github.com/Hildenus/VKR.git>. На него были загружены результаты работы: исследовательский notebook, код приложения.