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 Новак Дарья Александровна

Москва, 2023

СОДЕРЖАНИЕ

СОДЕРЖАНИЕ 2

ВВЕДЕНИЕ 3

1. АНАЛИТИЧЕСКАЯ ЧАСТЬ 4
   1. Постановка задачи 4
   2. Описание используемых методов. 4
   3. Разведочный анализ данных 9
2. ПРАКТИЧЕСКАЯ ЧАСТЬ 17
   1. Предобработка данных 17
   2. Разработка и обучение моделей машинного обучения. 18
   3. Нейронная сеть, которая будет рекомендовать соотношение

«матрица-наполнитель» 20

* 1. Разработка приложения 25

1. ЗАКЛЮЧЕНИЕ 28
2. СПИСОК ЛИТЕРАТУРЫ И ВЕБ РЕСУРСОВ 29

ВВЕДЕНИЕ

Композиционные материалы — это искусственно созданные материалы, состоящие из нескольких других с четкой границей между ними. Композиты обладают теми свойствами, которые не наблюдаются у компонентов по отдельности. При этом композиты являются монолитным материалом, т. е. компоненты материала неотделимы друг от друга без разрушения конструкции в целом. Яркий пример композита - железобетон. Бетон прекрасно сопротивляется сжатию, но плохо растяжению. Стальная арматура внутри бетона компенсирует его неспособность сопротивляться сжатию, формируя тем самым новые, уникальные свойства. Современные композиты изготавливаются из других материалов: полимеры, керамика, стеклянные и углеродные волокна, но данный принцип сохраняется. У такого подхода есть и недостаток: даже если мы знаем характеристики исходных компонентов, определить характеристики композита, состоящего из этих компонентов, достаточно проблематично.

Для решения этой проблемы есть два пути: физические испытания образцов материалов, или прогнозирование характеристик. Суть прогнозирования заключается в симуляции представительного элемента объема композита, на основе данных о характеристиках входящих компонентов (связующего и армирующего компонента). На входе имеются данные о начальных свойствах компонентов композиционных материалов (количество связующего, наполнителя, температурный режим отверждения и т.д.). На выходе необходимо спрогнозировать ряд конечных свойств получаемых композиционных материалов. Кейс основан на реальных производственных задачах Центра НТИ «Цифровое материаловедение: новые материалы и вещества» (структурное подразделение МГТУ им. Н.Э. Баумана). Актуальность: Созданные прогнозные модели помогут сократить количество проводимых испытаний, а также пополнить базу данных материалов возможными новыми характеристиками материалов, и цифровыми двойниками новых композитов.

* + 1. АНАЛИТИЧЕСКАЯ ЧАСТЬ
       1. Постановка задачи

Для исследовательской работы были даны 2 файла: X\_bp.xlsx (с данными о параметрах, состоящий из 1023 строк и 10 столбцов данных) и X\_nup.xlsx (данными нашивок, состоящий из 1040 строк и 3 столбцов данных).

Для разработки моделей по прогнозу модуля упругости при растяжении, прочности при растяжении и соотношения матрица-наполнитель нужно объединить 2 файла. Объединение по типу INNER, поэтому часть информации (17 строк таблицы X\_nup.xlsx) не имеет соответствующих строк в таблице X\_bp.xlsx и будет удалена.

Также необходимо провести разведочный анализ данных, нарисовать гистограммы распределения каждой из переменной, диаграммы boxplot (ящик с усами), попарные графики рассеяния точек.

Для каждой колонки получить среднее, медианное значение, провести анализ и исключение выбросов, проверить наличие пропусков; сделать предобработку: удалить шумы и выбросы, сделать нормализацию и стандартизацию.

Обучить несколько моделей для прогноза модуля упругости при растяжении и прочности при растяжении. Написать нейронную сеть, которая будет рекомендовать соотношение матрица-наполнитель. Разработать приложение с графическим интерфейсом, которое будет выдавать прогноз соотношения «матрица-наполнитель». Оценить точность модели на тренировочном и тестовом датасете. Создать репозиторий в GitHub и разместить код исследования. Оформить файл README.

* + - 1. 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 были применены следующие методы:

* линейная регрессия (Linear regression);
* лассо регрессия (Lasso);
* гребневая регрессия (Ridge);
* К-ближайших соседей (KNeighborsRegressor);
* случайный лес (RandomForest);
* градиентный бустинг (AdaBoostRegressor);
* полиноминальная регрессия (Polynominal regression)

***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.

Чтобы улучшить Линейную модель путем обмена некоторой этой дисперсии с предвзятостью, чтобы уменьшить нашу общую ошибку. Это происходит при помощи регуляризации, в которой модифицируется функция стоимости, чтобы ограничить значения коэффициентов. Это позволяет изменить чрезмерную дисперсию на некоторое смещение, потенциально уменьшая общую ошибку.

***Лассо регрессия (Lasso) –*** это линейная модель, которая оценивает разреженные коэффициенты. Это простой метод, позволяющий уменьшить сложность модели и предотвратить переопределение, которое может возникнуть в результате простой линейной регрессии. Данный метод вводит дополнительное слагаемое регуляризации в оптимизацию модели. Это даёт более устойчивое решение. В регрессии лассо добавляется условие смещения в функцию оптимизации для того, чтобы уменьшить коллинеарность и, следовательно, дисперсию модели. Но вместо квадратичного смещения, используется смещение абсолютного значения. Лассо регрессия хорошо прогнозирует модели временных рядов на основе регрессии, таким как авторегрессии.

*Достоинства метода:* легко полностью избавляется от шумов в данных; быстро работает; не очень энергоёмко; способно полностью убрать признак из датасета; доступно обнуляет значения коэффициентов.

*Недостатки метода:* часто страдает качество прогнозирования; выдаёт ложное срабатывание результата; случайным образом выбирает одну из коллинеарных переменных; не оценивает правильность формы взаимосвязи между независимой и зависимой переменными; не всегда лучше, чем пошаговая регрессия.

Лассо-регрессию следует использовать, когда есть несколько характеристик с высокой предсказательной способностью, а остальные бесполезны. Она обнуляет бесполезные характеристики и оставляет только подмножество переменных.

***Гребневая регрессия (Ridge) –*** это регрессия, которая добавляет дополнительный штраф к функции стоимости, но вместо этого суммирует квадраты значений коэффициентов (норма L-2) и умножает их на некоторую постоянную лямбду. По сравнению с Лассо этот штраф регуляризации уменьшит значения коэффициентов, но не сможет принудительно установить коэффициент равным 0. Это ограничивает использование регрессии гребня в отношении выбора признаков. Однако, когда p> n, он способен выбрать более n релевантных предикторов, если необходимо, в отличие от Лассо. Он также выберет группы коллинеарных элементов, которые его изобретатели назвали «эффектом группировки».

Как и в случае с Лассо, мы можем варьировать лямбду, чтобы получить модели с различными уровнями регуляризации, где лямбда = 0 соответствует OLS, а лямбда приближается к бесконечности, что соответствует постоянной функции.

Анализ регрессии Лассо, так и Риджа показывает, что ни один метод не всегда лучше, чем другой; нужно попробовать оба метода, чтобы определить, какой использовать.

Ридж-регрессию лучше применять, когда предсказательная способность набора данных распределена между различными характеристиками. Ридж- регрессия не обнуляет характеристики, которые могут быть полезны при составлении прогнозов, а просто уменьшает вес большинства переменных в модели.

***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 трудоемкость.

***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 вместе.

*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 недообучаться; трудоёмко прогнозируемый; иногда работает хуже, чем линейные методы.

***Полиноминальная регрессия (Polynominal regression) -*** это алгоритм машинного обучения, который используется для обучения линейной модели на нелинейных данных. Чтобы реализовать этот алгоритм, сначала необходимо обучить линейную модель на нелинейном наборе данных, скажем, мы хотим обучить алгоритм линейной регрессиина нелинейных данных, что даст нам плохие результаты. Поэтому в таких случаях нужно добавить новые функции, учитывая существующие функции, чтобы сделать наш набор данных линейным. Для этой задачи нам нужно использовать алгоритм полиномиальной регрессии, который преобразует данные в расширенные, а затем, если мы снова будем использовать алгоритм линейной регрессии для вновь преобразованных данных, он даст нам идеальные результаты.

*Достоинства метода:* Моделирует нелинейно разделенные данные (чего не может линейная регрессия). Она более гибкая и может моделировать сложные взаимосвязи. Полный контроль над моделированием переменных объекта (выбор степени).

*Недостатки метода:* Необходимо внимательно создавать модель. Необходимо обладать некоторыми знаниями о данных, для выбора наиболее подходящей степени. При неправильном выборе степени, данная модель может быть перенасыщена.

**Используемые метрики качества моделей:**

***R2*** -коэффициент детерминации***,*** измеряет долю дисперсии, объяснённую моделью, вобщей 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.

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

***MAE (Mean Absolute Error)*** - это усреднённая сумма модулей разницы между реальным и предсказанным значениями. MAE во многом похожа на MSE, но она отличается меньшей чувствительностью к выбросам значений (так как не берётся квадрат отклонения).

* + - 1. Разведочный анализ данных

Прежде чем передать данные в работу моделей машинного обучения, необходимо обработать и очистить их. Необработанные данные могут содержать искажения и пропущенные значения и способны привести к неверным результатам по итогам моделирования. Но безосновательно удалять что-либо тоже неправильно. Именно поэтому сначала набор данных надо изучить.

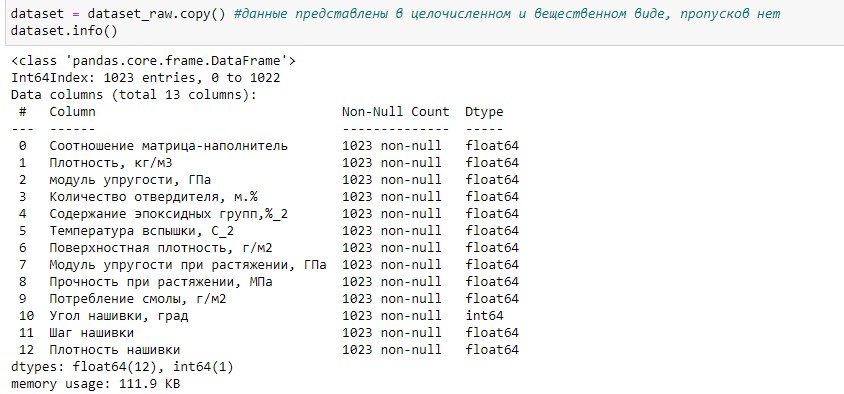


Рисунок 1 – Общая информация по датасету.

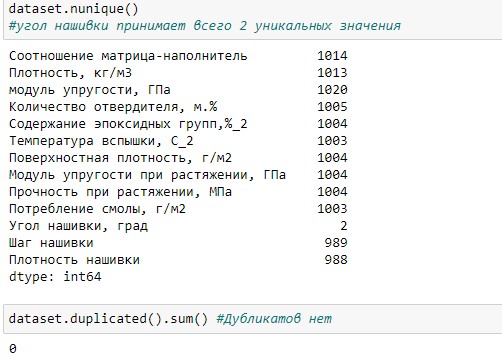


Рисунок 2 – Уникальные значения и дубликаты.

Цель разведочного анализа - получение первоначальных представлений о характерах распределений переменных исходного набора данных, формирование оценки качества исходных данных (наличие пропусков, выбросов), выявление характера взаимосвязи между переменными с целью последующего выдвижения гипотез. В качестве инструментов разведочного анализа используется: оценка статистических характеристик датасета; гистограммы распределения каждой из переменной; диаграммы boxplot (ящика с усами); попарные графики рассеяния точек; тепловая карта; описательная статистика для каждой переменной; анализ и полное исключение выбросов; проверка наличия пропусков и дубликатов; корреляция Кендалла и Пирсона.

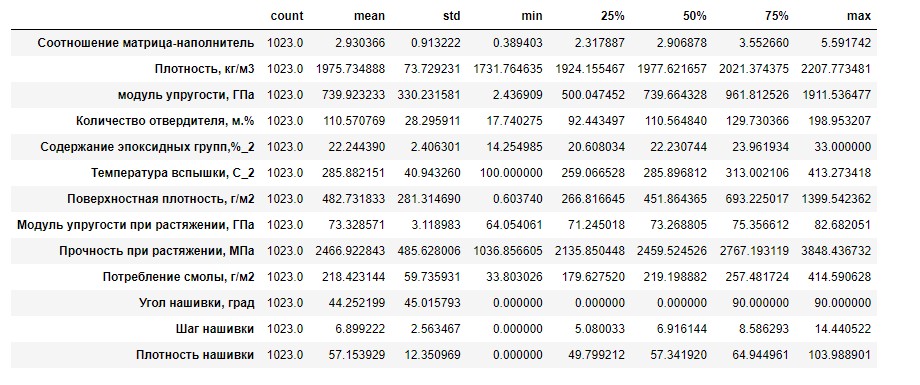


Рисунок 3 – Описательная статистика датасета

Описательная статистика содержит по каждой колонке:

- count - количество значений

- mean - среднее значение

- std - стандартное отклонение

- min - минимум

- 25% - верхнее значение первого квартиля

- 50% - медиана

- 75% - верхнее значение третьего квартиля

- max - максимум

Для анализа описательной статистики необходимо привлечение специалиста.

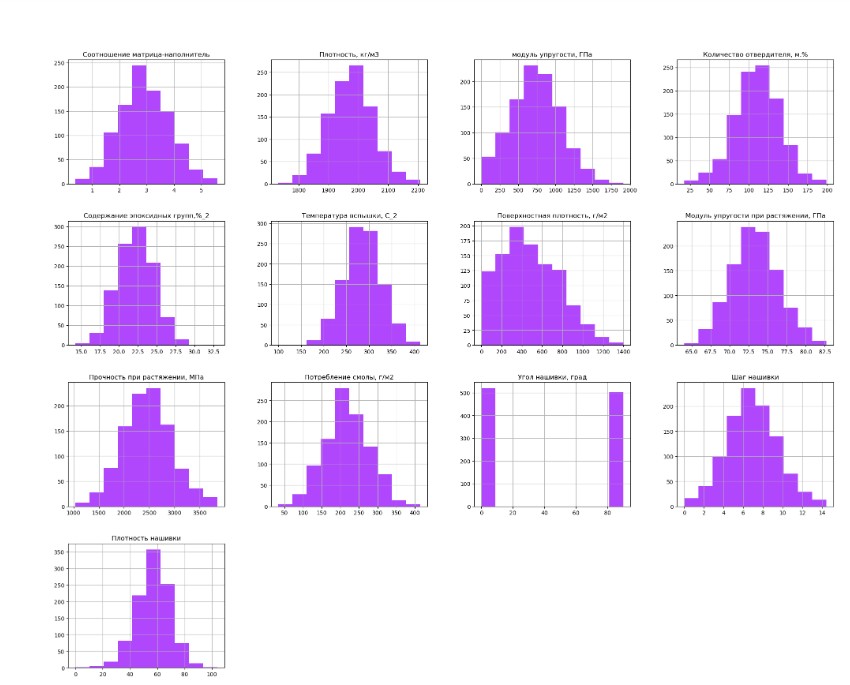


Рисунок 4 – Гистограммы распределения переменных

Большинство признаков имеют нормальное распределение, присутствуют выбросы. Поверхностная плотность имеет положительную ассиметрию –ассиметричное распределение со смещением влево, длинный правый хвост. Наблюдается бинарность признаков угла нашивки.

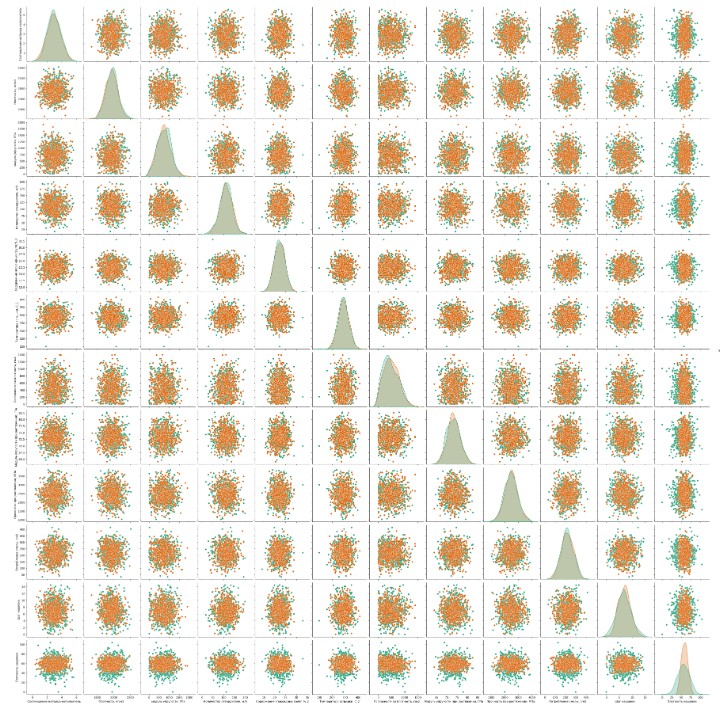


Рисунок 5 – Попарные графики рассеяния с выделением значений Угол нашивки

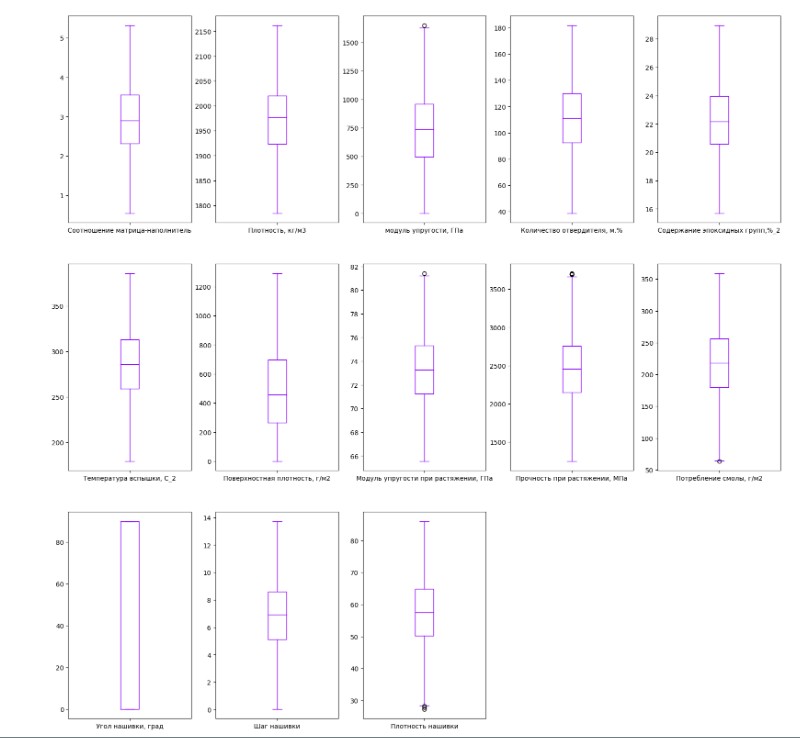


Рисунок 6 – визуализация выбросов до обработки.

Одним из самых больших недостатков для производительности любой модели являются выбросы, присутствующие в данных. В идеале выбросы - это экстремальные значения для определенного столбца, которые влияют на обобщение данных и модели. Выбросы в основном влияют на модели регрессии, поскольку они радикально меняют уравнение. Цель обнаружения и обработки выбросов - гарантировать, что вы получите наилучшую модель из данных, учитывая тот факт, что ваши данные подходят для работы с алгоритмом.

Для удаления выбросов используем метод межквартильного расстояния для максимальной чистоты, так как используем методы чувствительные к выбросам.

Межквартильное расстояние - определяется как разница между верхним и нижним квартилями в наборе данных. Квартили - это значения, которые разделяют упорядоченный набор данных на четыре равные части. Верхний квартиль (Q3) - это значение, которое делит верхние 25% данных от нижних 75%, а нижний квартиль (Q1) - это значение, которое делит нижние 25% данных от верхних 75%. Таким образом, Межквартильное расстояние (IQR) - это разница между Q3 и Q1. Он представляет собой интерквартильный диапазон, который содержит 50% данных и позволяет оценить разброс значений в наборе данных, игнорируя выбросы.

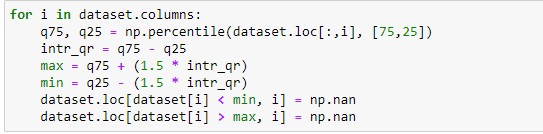


Рисунок 7 – визуализация выбросов до обработки.

После обработки было удалено 87 строк из датасета.

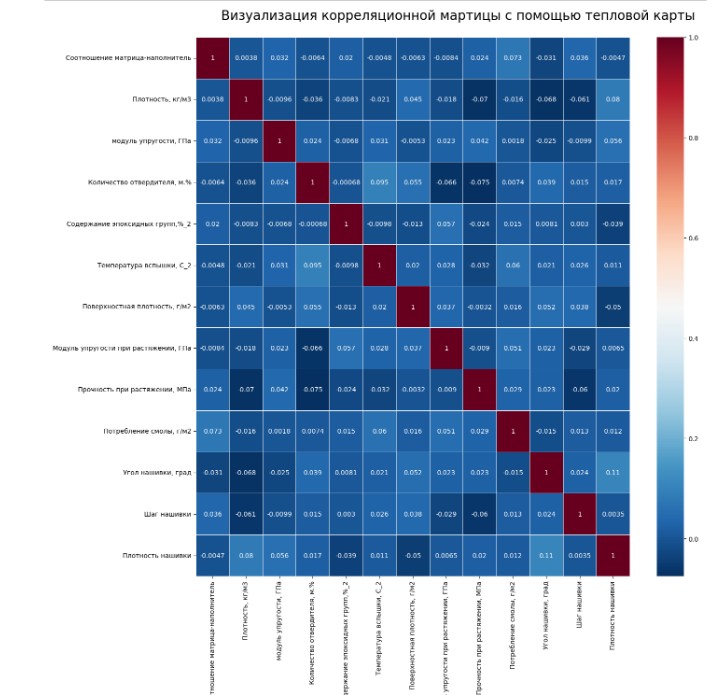


Рисунок 8 – Тепловая карта с корреляцией данных

Тепловая карта показывает корреляцию близкую к нулю, что означает отсутствие линейной зависимости между переменными.

* + 1. ПРАКТИЧЕСКАЯ ЧАСТЬ
       1. Предобработка данных

По условиям задания нормализуем значения. Для этого применим MinMaxScaler.

Масштабирование данных - это этап предварительной обработки данных для числовых объектов. Многие алгоритмы машинного обучения, такие как методы градиентного спуска, алгоритм KNN, линейная и логистическая регрессия и т.д., Требуют масштабирования данных для получения хороших результатов. Для этой цели определены различные масштабаторы.

MinMax Scaler приводит данные к заданному диапазону (по умолчанию к промежутку от 0 до 1), сохраняет форму исходного распределения. Он не меняет содержательно информацию содержащуюся в исходных данных.

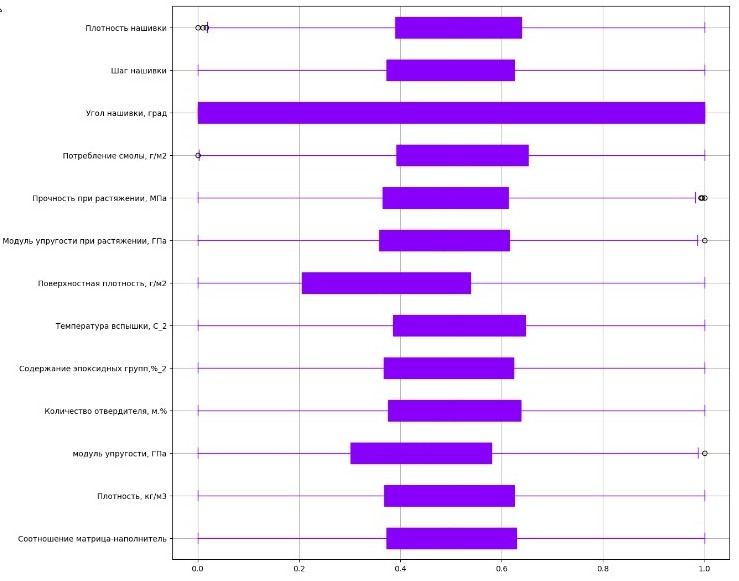


Рисунок 9 – Масштабированные данные

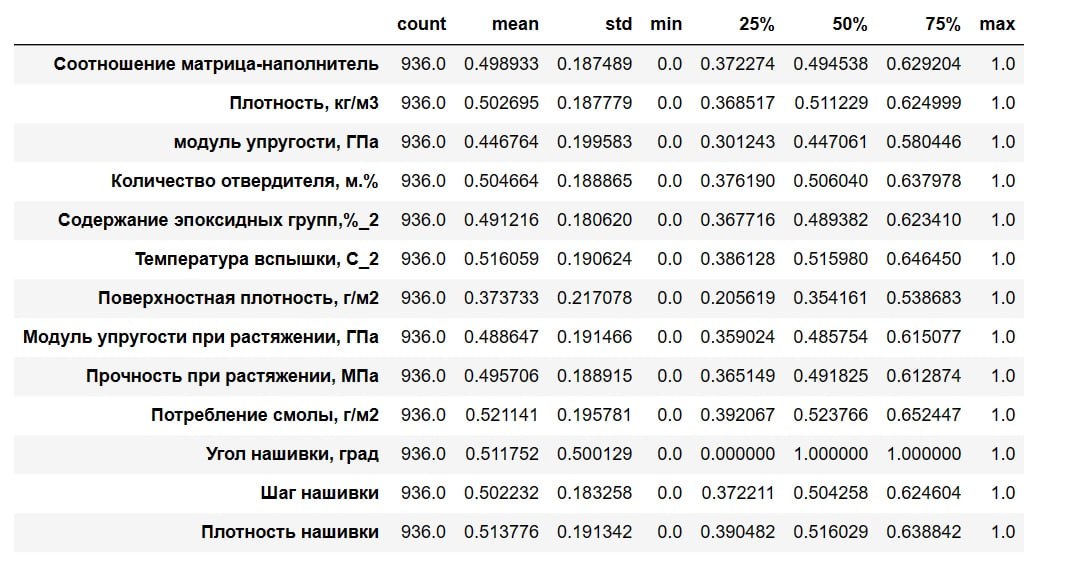


Рисунок 10 – Описательная статистика масштабированных данных

* + - 1. Разработка и обучение моделей машинного обучения

Предсказание значений вещественной, непрерывной переменной — это задача регрессии. Эта зависимая переменная должна иметь связь с одной или несколькими независимыми переменными, называемых также предикторами или регрессорами. Регрессионный анализ помогает понять, как «типичное» значение зависимой переменной изменяется при изменении независимых переменных.

В настоящее время разработано много методов регрессионного анализа. Например, простая и множественная линейная регрессия. Эти модели являются параметрическими в том смысле, что функция регрессии определяется конечным числом неизвестных параметров, которые оцениваются на основе данных.

Разработка и обучение моделей машинного обучения осуществлялась для двух выходных параметров: «Прочность при растяжении» и «Модуль упругости при растяжении» отдельно. Порядок разработки модели для каждого параметра и для каждого выбранного метода можно разделить на следующие этапы: разделение нормализованных данных на обучающую и тестовую выборки (в соотношении 70 на 30%); обучение моделей на нормализованных значениях; поиск гиперпараметров, по которым будет происходить оптимизация модели, с помощью выбора по сетке и перекрёстной проверки. Оценка полученных результатов работы моделей. В качестве параметра оценки выбран также коэффициент детерминации (R2).

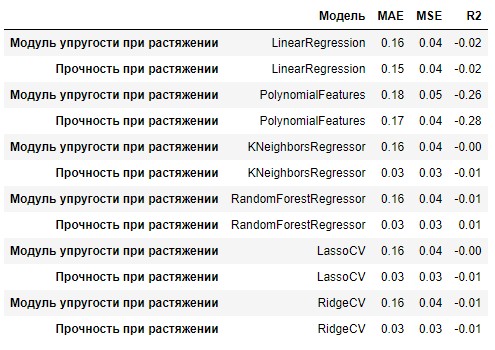


Рисунок 11 – Метрики моделей обучения

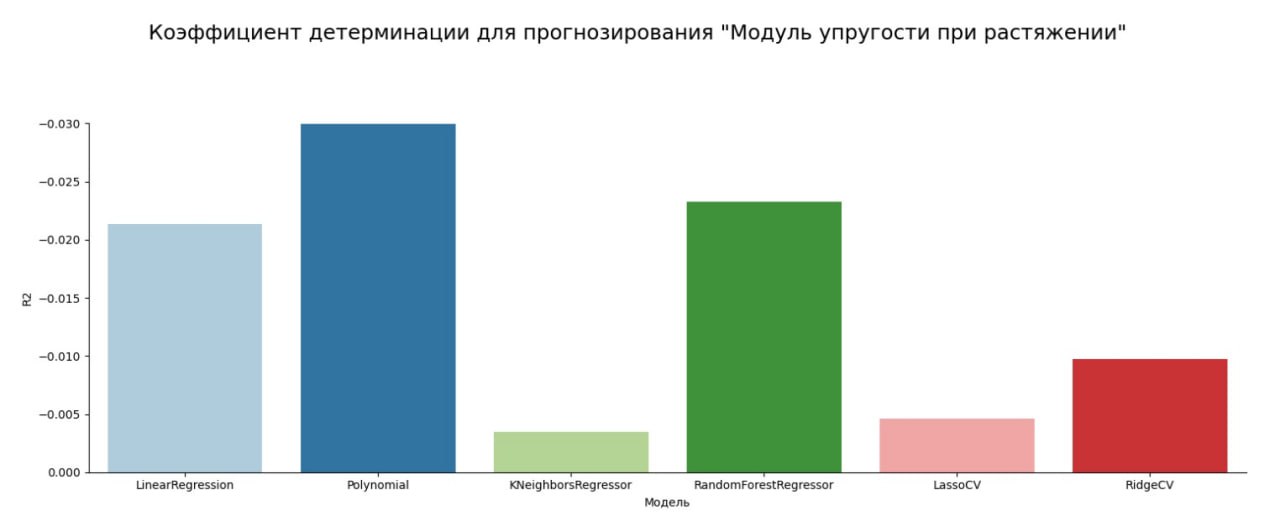


Рисунок 12 – R2 моделей обучения

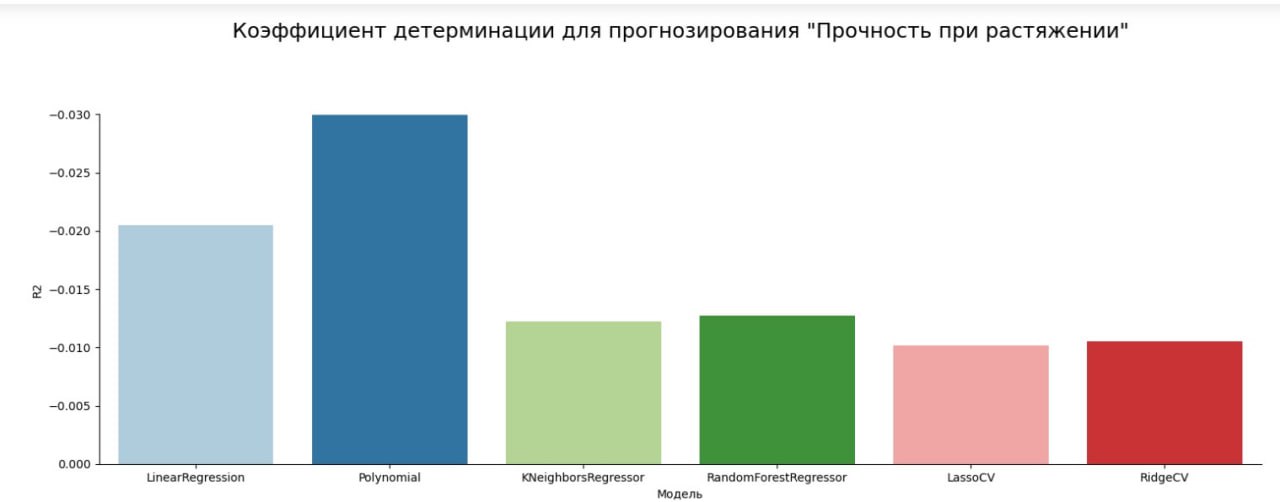


Рисунок 13 – R2 моделей обучения

В результате все модели показали примерно одинаковый результат: ошибка MAE примерно равна стандартному отклонению, значения R2 находятся около нуля, то есть все модели предсказывают результат сопоставимый со средним значением. Можно считать, что все примененные модели не справились с задачей, результат неудовлетворительный.

* 1. Нейронная сеть для рекомендации «Соотношение матрица-наполнитель».

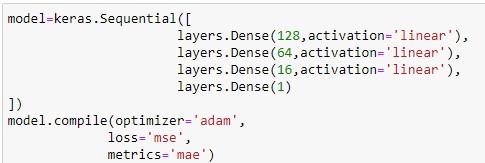
Искусственная нейронная сеть (ИНС) (англ. Artificial neural network (ANN)) — упрощенная модель биологической нейронной сети, представляющая собой совокупность искусственных нейронов, взаимодействующих между собой.

Основные принципы работы нейронных сетей были описаны еще в 1943 году Уорреном Мак-Каллоком и Уолтером Питтсом. В 1957 году нейрофизиолог Фрэнк Розенблатт разработал первую нейронную сеть, а в 2010 году большие объемы данных для обучения открыли возможность использовать нейронные сети для машинного обучения.

На данный момент нейронные сети используются в многочисленных областях машинного обучения и решают проблемы различной сложности.

Хорошим примером биологической нейронной сети является человеческий мозг. Наш мозг — сложнейшая биологическая нейронная сеть, которая принимает информацию от органов чувств и каким-то образом ее обрабатывает (узнавание лиц, возникновение ощущений и т.д.). Мозг же, в свою очередь, состоит из нейронов, взаимодействующих между собой.

Для построения искусственной нейронной сети будем использовать ту же структуру. Как и биологическая нейронная сеть, искусственная состоит из нейронов, взаимодействующих между собой, однако представляет собой упрощенную модель. Так, например, искусственный нейрон, из которых состоит ИНС, имеет намного более простую структуру: у него есть несколько входов, на которых он принимает различные сигналы, преобразует их и передает другим нейронам. Другими словами, искусственный нейрон — это такая функция Rn→R, которая преобразует несколько входных параметров в один выходной. Для рекомендации «Соотношение матрица-наполнитель %» была построена многослойная полносвязная нейронная сеть.

  
Рисунок 14 – Программный код нейронной сети

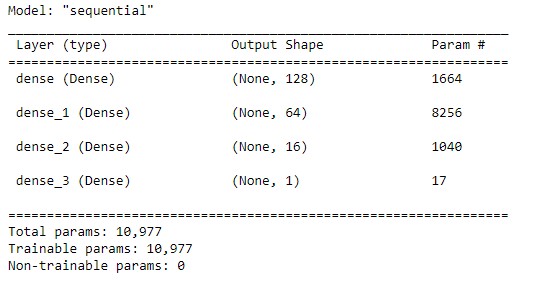


Рисунок 15 – Структура нейронной сети.

Поведем обучение нейронной сети и проверку работы на тестовой выборке.



Рисунок 16 – Метрики нейронной сети.

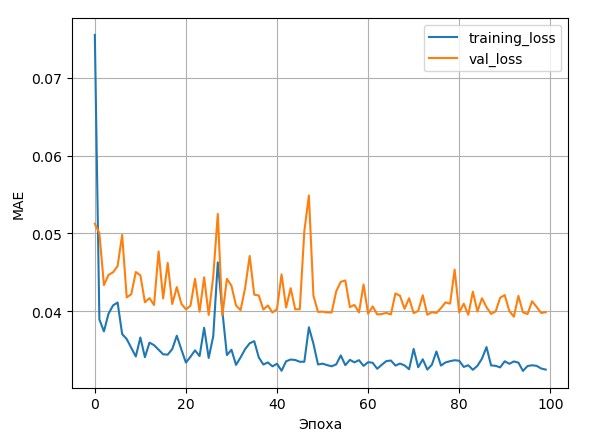


Рисунок 17 – График изменения средней абсолютной ошибки

при обучении модели.

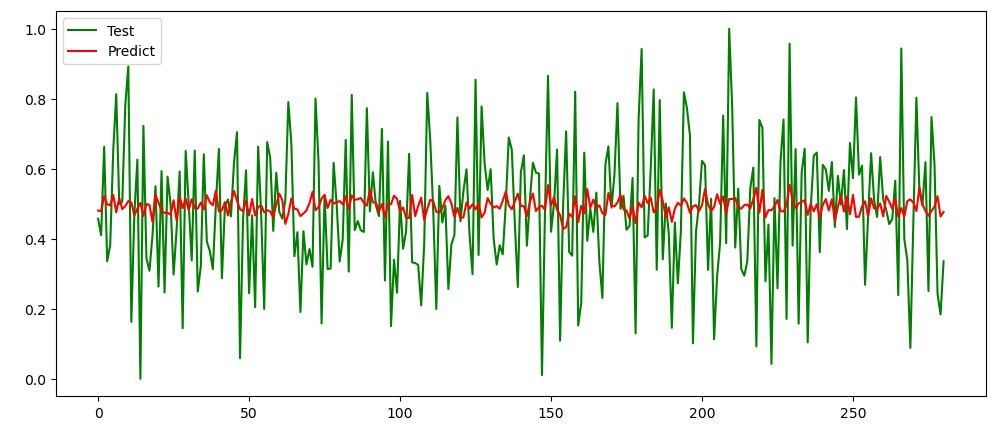


Рисунок 18 – Тестовые и прогнозные значения.

Таким образом, построенная модель неэффективна, и не решает поставленную задачу. Попробуем построить другие модели, поэкспериментировав с параметрами.

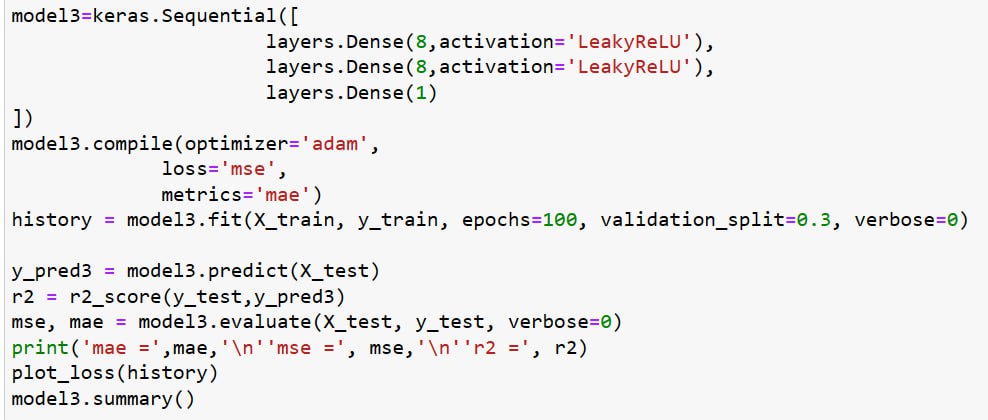


Рисунок 19 – Программный код нейронной сети

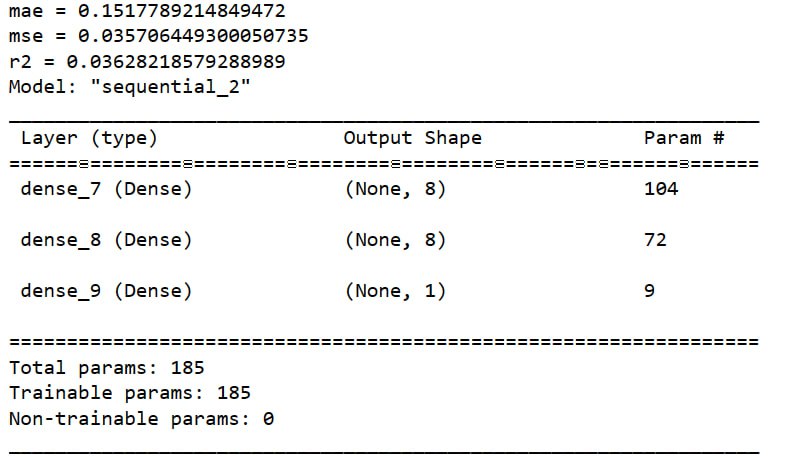


Рисунок 20 – Метрики и архитектура нейронной сети.

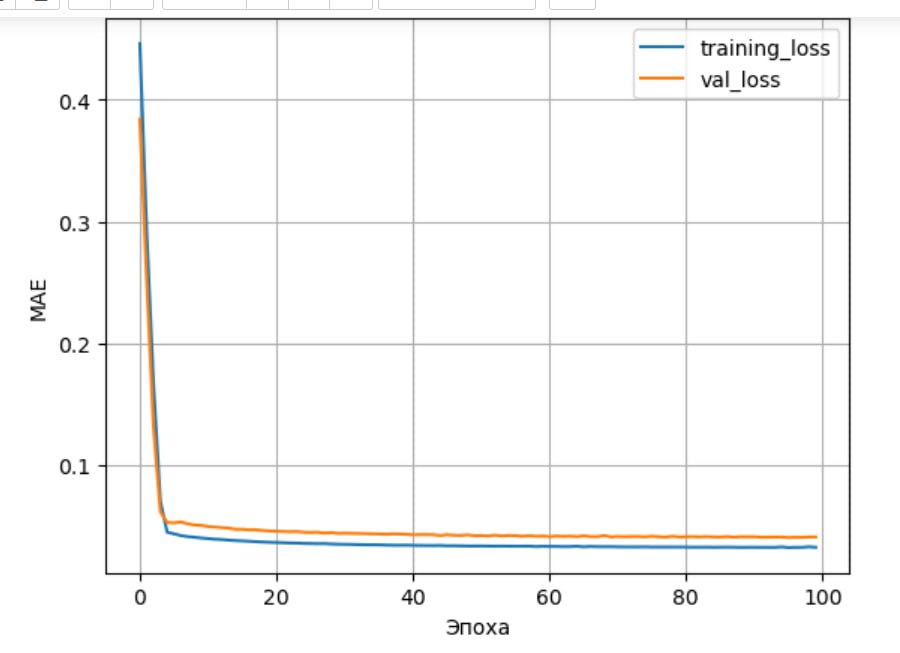


Рисунок 21 – График изменения средней абсолютной ошибки

при обучении модели.

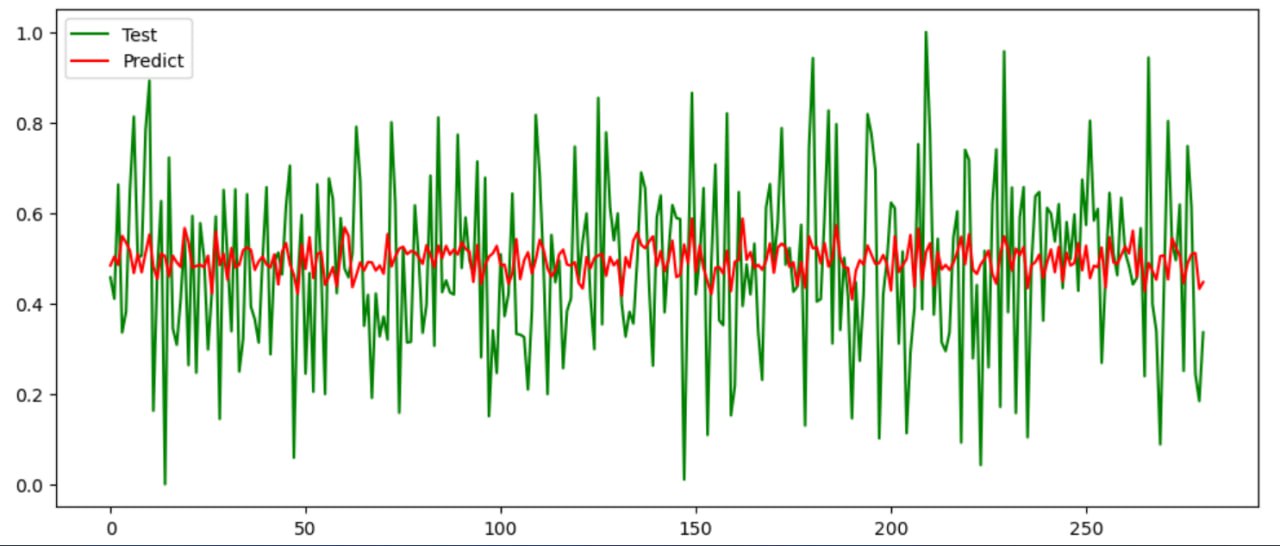


Рисунок 22 – Тестовые и прогнозные значения.

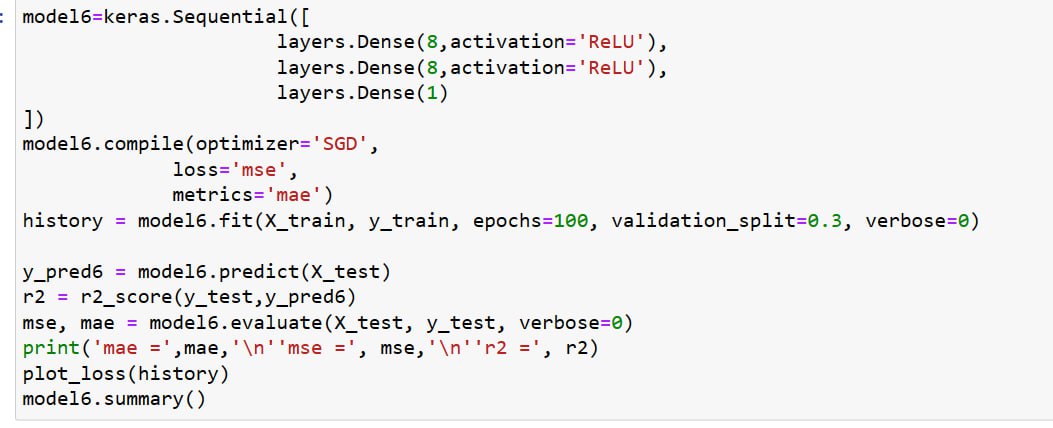


Рисунок 23 – Программный код нейронной сети

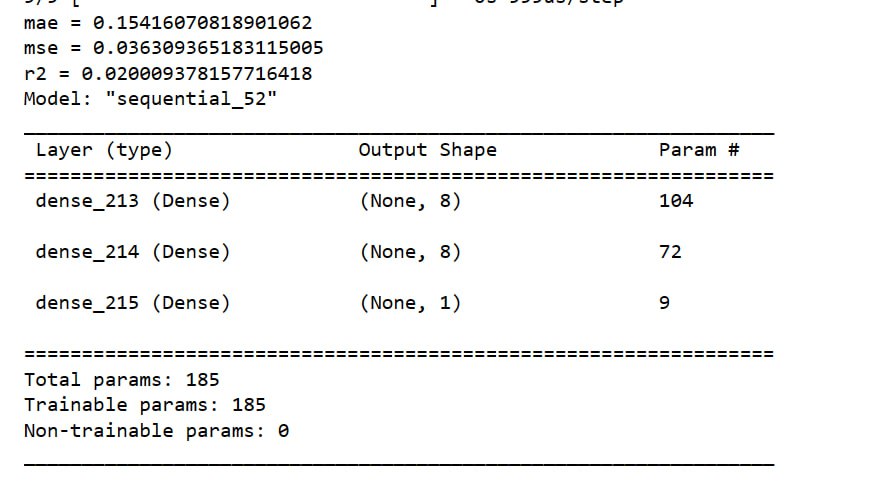


Рисунок 24 – Метрики и архитектура нейронной сети.

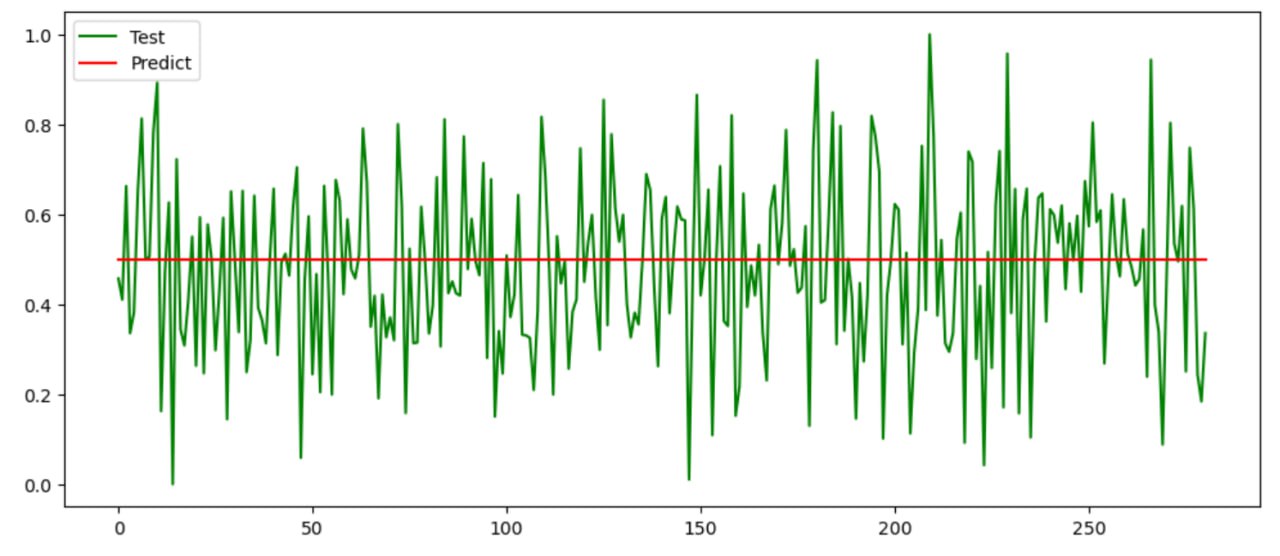


Рисунок 25 – Тестовые и прогнозные значения.

* 1. Разработка приложения.

Создание приложения для расчета параметра «Соотношение матрица- наполнитель». Данное приложение разработано с помощью микро-фреймворка Flask.

Flask — это легковесный веб-фреймворк для языка Python, который предоставляет минимальный набор инструментов для создания веб-приложений.

У Flask есть ряд особенностей, за которые его любят веб-разработчики. Перечислим их :

* Минимальный набор инструментов из коробки. Причём они не навязывают какую-то архитектуру или жёсткую структуру проектов. Разработчики сами решают, как и что они будут создавать.
* Гибкость. Работая с Flask, программист может выбирать только необходимые встроенные инструменты и подключать дополнительные внешние, не перегружая проект лишними модулями.
* Расширяемость. У Flask много расширений и плагинов, которые помогают быстро добавить новую функциональность. Например, авторизацию, управление базами данных и работу с формами.
* Простота. У Flask простой синтаксис, что делает изучение этого фреймворка более простым, а также позволяет быстрее создавать прототипы веб-приложений.
* Поддержка сообщества. Flask запустили в 2010 году, и почти по любому связанному с ним вопросу в интернете уже есть ответы.

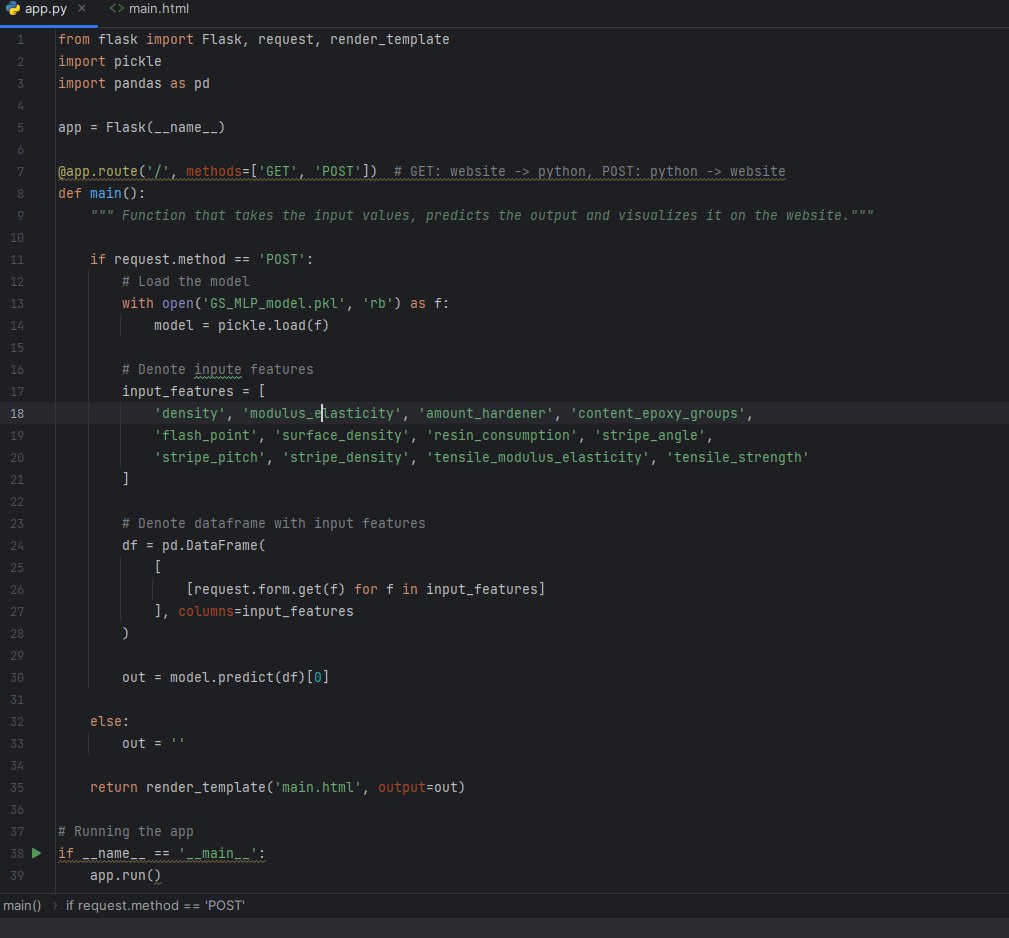


Рисунок 26 – Код приложения.

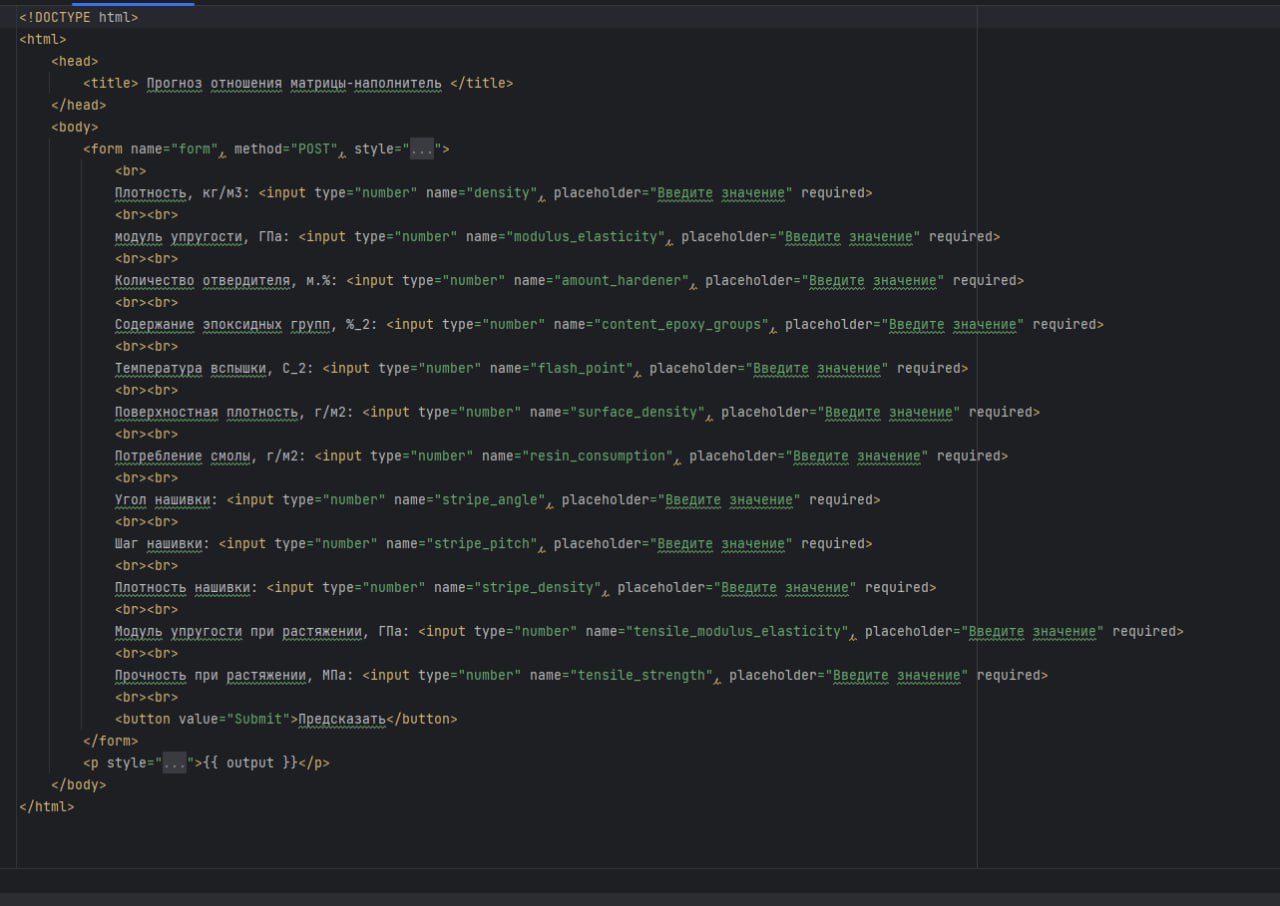


Рисунок 27 – HTML.

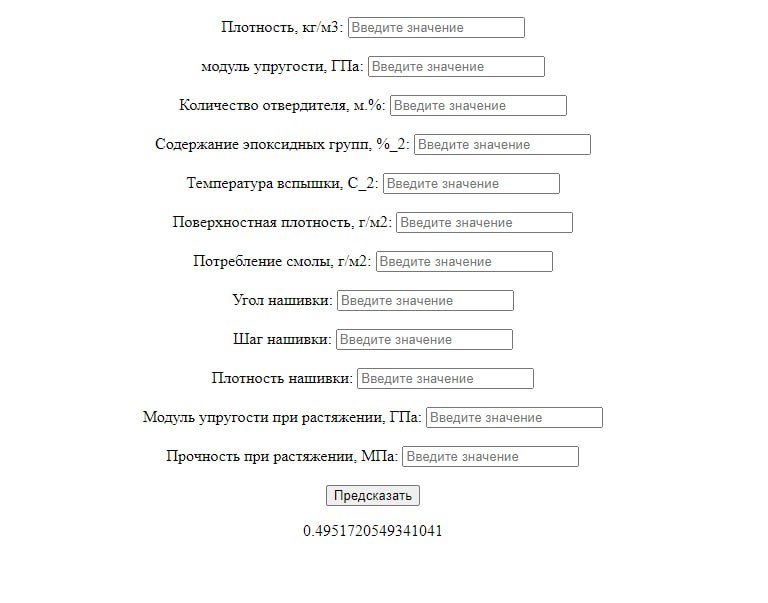


Рисунок 27 – Реализация приложения.

Репозиторий был создан на github.com по адресу : <https://github.com/NowakDA/VKR>

1. ЗАКЛЮЧЕНИЕ

Данная исследовательская работа позволяет сделать некоторые основные выводы по теме. Распределение полученных данных в объединённом датасете близко к нормальному, корреляция между парами признаков близка к нулю. Использованные при разработке моделей подходы не позволили получить достоверных прогнозов. Применённые модели регрессии не показали эффективности в прогнозировании свойств композитов.

Можно сделать 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. В целом прогнозирование конечных свойств/характеристик композитных материалов без изучения материаловедения, погружения в вопрос экспериментального анализа характеристик композитных материалов не демонстрирует удовлетворительных результатов. Учитывая отсутствие корреляции между признаками, делаем вывод, что текущим набором алгоритмов задача не решается, возможно, решается трудно или не решается совсем.

1. СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ И ВЕБ-РЕСУРСЫ
2. Alex Maszański. Метод k-ближайших соседей (k-nearest neighbour) https://proglib.io/p/metod-k-blizhayshih-sosedey-k-nearest-neighbour2021-07-19.
3. Andre Ye. 5 алгоритмов регрессии в машинном обучении, о которых вам следует знать: https://habr.com/ru/company/vk/blog/513842/
4. [Anthony Schams](https://medium.com/%40acschams?source=post_page-----8bf81991d0c5--------------------------------). Смещение, дисперсия и регуляризация в линейной регрессии: лассо, хребет и эластичная сеть - различия и использовании https://machinelearningmastery.ru/bias-variance-and-regularization-in-linear- regression-lasso-ridge-and-elastic-net-8bf81991d0c5/
5. Devpractice Team. Python. Визуализация данных. Matplotlib. Seaborn. Mayavi. - devpractice.ru. 2020. - 412 с.: ил.
6. Абросимов Н.А Методика построения разрешающей системы уравнений динамического деформирования композитных элементов конструкций (Учебно-методическое пособие), ННГУ, 2010
7. Абу-Хасан Махмуд, Масленникова Л. Л. Прогнозирование свойств композиционных материалов с учётом наноразмера частиц и акцепторных свойств катионов твёрдых фаз, статья 2006 год
8. Гафаров, Ф.М., Галимянов А.Ф. Искусственные нейронные сети и приложения: учеб. пособие /Ф.М. Гафаров, А.Ф. Галимянов. – Казань: Издательство Казанского университета, 2018. – 121 с.
9. Грас Д. Data Science. Наука о данных с нуля: Пер. с англ. - 2-е изд., перераб. и доп. - СПб.: БХВ-Петербурr, 2021. - 416 с.: ил.9.
10. Документация по библиотеке keras: https://keras.io/api/.
11. Документация по библиотеке matplotlib: https://matplotlib.org/stable/users/index.html.
12. Документация по библиотеке numpy: https://numpy.org/doc/1.22/user/index.html#user.
13. Документация по библиотеке pandas: https://pandas.pydata.org/docs/user\_guide/index.html#user-guide.
14. Документация по библиотеке scikit-learn: https://scikit-learn.org/stable/user\_guide.html.
15. Документация по библиотеке seaborn: https://seaborn.pydata.org/tutorial.html.
16. Документация по библиотеке Tensorflow: https://[www.tensorflow.org/overview](http://www.tensorflow.org/overview)
17. Документация по языку программирования python: https://docs.python.org/3.8/index.html.
18. Иванов Д.А., Ситников А.И., Шляпин С.Д – Композиционные материалы: учебное пособие для вузов, 2019. 13 с.
19. Краткий обзор алгоритма машинного обучения Метод Опорных Векторов (SVM) : https://habr.com/ru/post/428503/
20. Ларин А. А., Способы оценки работоспособности изделий из композиционных материалов методом компьютерной томографии, Москва, 2013, 148 с.
21. [Макс](https://your-scorpion.ru/) Ветков. Градиентный бустинг(AdaBoost) https://your- scorpion.ru/gradient-boosting-adaboost
22. Материалы конференции: V Всероссийская научно-техническая конференция «Полимерные композиционные материалы и производственные технологии нового поколения», 19 ноября 2021 г.
23. Плас Дж. Вандер, Python для сложных задач: наука о данных и машинное обучение. Санкт-Петербург: Питер, 2018, 576 с
24. Реутов Ю.А.: Прогнозирование свойств полимерных композиционных материалов и оценка надёжности изделий из них, Диссертация на соискание учёной степени кандидата физико-математических наук, Томск 2016.
25. Роббинс, Дженнифер. HTML5: карманный справочник, 5-е издание.: Пер. с англ. - М.: ООО «И.Д. Вильямс»: 2015. - 192 с.: ил
26. Руководство по быстрому старту в flask: – Режим доступа: https://flaskrussian-docs.readthedocs.io/ru/latest/quickstart.html. (дата обращения: 09.06.2022)