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 PRO»

Слушатель Виноградов Максим Александрович

Москва, 2023

Содержание

Введение

Данная работа выполнена в рамках курса Data Science PRO.

В качестве анализируемой задачи принята тема «Прогнозирование конечных свойств новых материалов (композиционных материалов).

Композиционные материалы — это многокомпонентные материалы, изготовленные из двух или более компонентов с существенно различными физическими и/или химическими свойствами, которые, в сочетании, приводят к появлению нового материала с характеристиками, отличными от характеристик отдельных компонентов.

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).

При этом существует проблема определения свойств нового материала даже при известных свойствах его составляющих.

Для решения этой проблемы есть два пути: физические испытания образцов материалов, или прогнозирование характеристик на основе данных о характеристиках входящих компонентов (связующего и армирующего компонента).

В работе разрабатываются модели прогнозирования ряда конечных свойств композиционного материала на основе данных реальных производственных задачач Центра НТИ «Цифровое материаловедение: новые материалы и вещества» (структурное подразделение МГТУ им. Н.Э. Баумана).

1 Аналитическая часть

1.1 Постановка задачи

Тема: Прогнозирование конечных свойств новых материалов (композиционных материалов).

Кейс основан на реальных производственных задачах Центра НТИ «Цифровое материаловедение: новые материалы и вещества» (структурное подразделение МГТУ им. Н.Э. Баумана).

На входе имеются данные о начальных свойствах компонентов композиционных материалов. На выходе необходимо спрогнозировать ряд конечных свойств получаемых композиционных материалов: модуль упругости при растяжении, прочность при растяжении и соотношение матрица-наполнитель.

Разработка модели для прогноза конечных свойств композиционного материала направлена на сокращение количества проводимых испытаний, а также пополнить базу данных материалов возможными новыми характеристиками материалов, и цифровыми двойниками новых композитов.

В качестве датасета имеются две таблицы формата .xlsx: X_bp.xlsx, состоящий из 1024 строк и 10 столбцов и X_nup.xlsx, состоящий из 1041 строки и 3 столбцов (индексный столбец не учитывается)

Датасет содержит в себе данные о свойствах исходных компонентов и итоговых композиционных материалов:

- Соотношение матрица-наполнитель;
- Плотность;
- Модуль упругости;
- Количество отвердителя;
- Содержание эпоксидных групп;
- Температура вспышки;
- Поверхностная плотность;
- Модуль упругости при растяжении;
- Прочность при растяжении;
- Потребление смолы;
- Угол нашивки;
- Шаг нашивки;
- Плотность нашивки.

Общее количество параметров для анализа – 13.

Разведочный анализ датасета показал наличие выбросов в данных, пропуски отсутствуют. Объединение датасета происходило по типу INNER. Часть информации (17 строк таблицы способов компоновки композитов) не имеющая соответствующих строк в таблице соотношений и свойств используемых компонентов композитов, поэтому были удалены.

Итоговый датасет нормализован, выбросы удалены, элементы массива соответствуют типу float64. Общий размер итоговой выборки - 922 строк, 13 столбцов.

1.2 Описание используемых методов

Для решения поставленной задачи выбран метод решения задачи регрессии для прогнозирования параметров: модуля упругости и прочности при растяжении. Для решения задачи регрессии использовались:

- линейная регрессия
- случайный лес
- метод k-ближайших соседей
- метод опорных векторов
- метод градиентного бустинга

Линейная регрессия (Linear regression) — это алгоритм машинного обученияс учителем, 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 R^2, 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 R^2 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.

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.

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.

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.

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.

Сравнение результатов методов приводится в пункте 2.2.

1.3 Разведочный анализ данных

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 о наиболее подходящих для решения задачи моделях машинного обучения.

Для разведочного анализа данных использованы методы описательной статистики.

При оценке числа уникальных значений датасета (DataFrame.nunique()) выявлено, что параметр «Угол нашивки» имеет дискретный характер и принимает только два значения: 0 и 90 град.

На рисунке 1 приведена полученная командой DataFrame.describe() описательная статистика датасета:

- среднее
- стандартное отклонение
- минимальное и максимальное значения
- 25, 50 и 75 процентили

	count	mean	std	min	25%	50%	75%	max
Соотношение матрица-наполнитель	1023.0	2.930366	0.913222	0.389403	2.317887	2.906878	3.552660	5.591742
Плотность, кг/м3	1023.0	1975.734888	73.729231	1731.764635	1924.155467	1977.621657	2021.374375	2207.773481
модуль упругости, ГПа	1023.0	739.923233	330.231581	2.436909	500.047452	739.664328	961.812526	1911.536477
Количество отвердителя, м.%	1023.0	110.570769	28.295911	17.740275	92.443497	110.564840	129.730366	198.953207
Содержание эпоксидных групп,%_2	1023.0	22.244390	2.406301	14.254985	20.608034	22.230744	23.961934	33.000000
Температура вспышки, С_2	1023.0	285.882151	40.943260	100.000000	259.066528	285.896812	313.002106	413.273418
Поверхностная плотность, г/м2	1023.0	482.731833	281.314690	0.603740	266.816645	451.864365	693.225017	1399.542362
Модуль упругости при растяжении, ГПа	1023.0	73.328571	3.118983	64.054061	71.245018	73.268805	75.356612	82.682051
Прочность при растяжении, МПа	1023.0	2466.922843	485.628006	1036.856605	2135.850448	2459.524526	2767.193119	3848.436732
Потребление смолы, г/м2	1023.0	218.423144	59.735931	33.803026	179.627520	219.198882	257.481724	414.590628
Угол нашивки, град	1023.0	44.252199	45.015793	0.000000	0.000000	0.000000	90.000000	90.000000
Шаг нашивки	1023.0	6.899222	2.563467	0.000000	5.080033	6.916144	8.586293	14.440522
Плотность нашивки	1023.0	57.153929	12.350969	0.000000	49.799212	57.341920	64.944961	103.988901

Рисунок 1 – описательная статистика датасета

Для проверки датасета наличие пропусков в значениях использованы команды df.isna() и df.isnull()). Пропуски в датасете отсутствуют. Дубликаты строк (DataFrame.duplicated()) также отсутствуют, все строки датасета уникальны.

Оценка распределений величин датасета проведена при помощи построения гистограмм распределения для каждой величины. Для построения использована библиотека визуализации встроенная функция библиотеки Pandas DataFrame.hist(). Гистограммы приведены на рисунке 2.

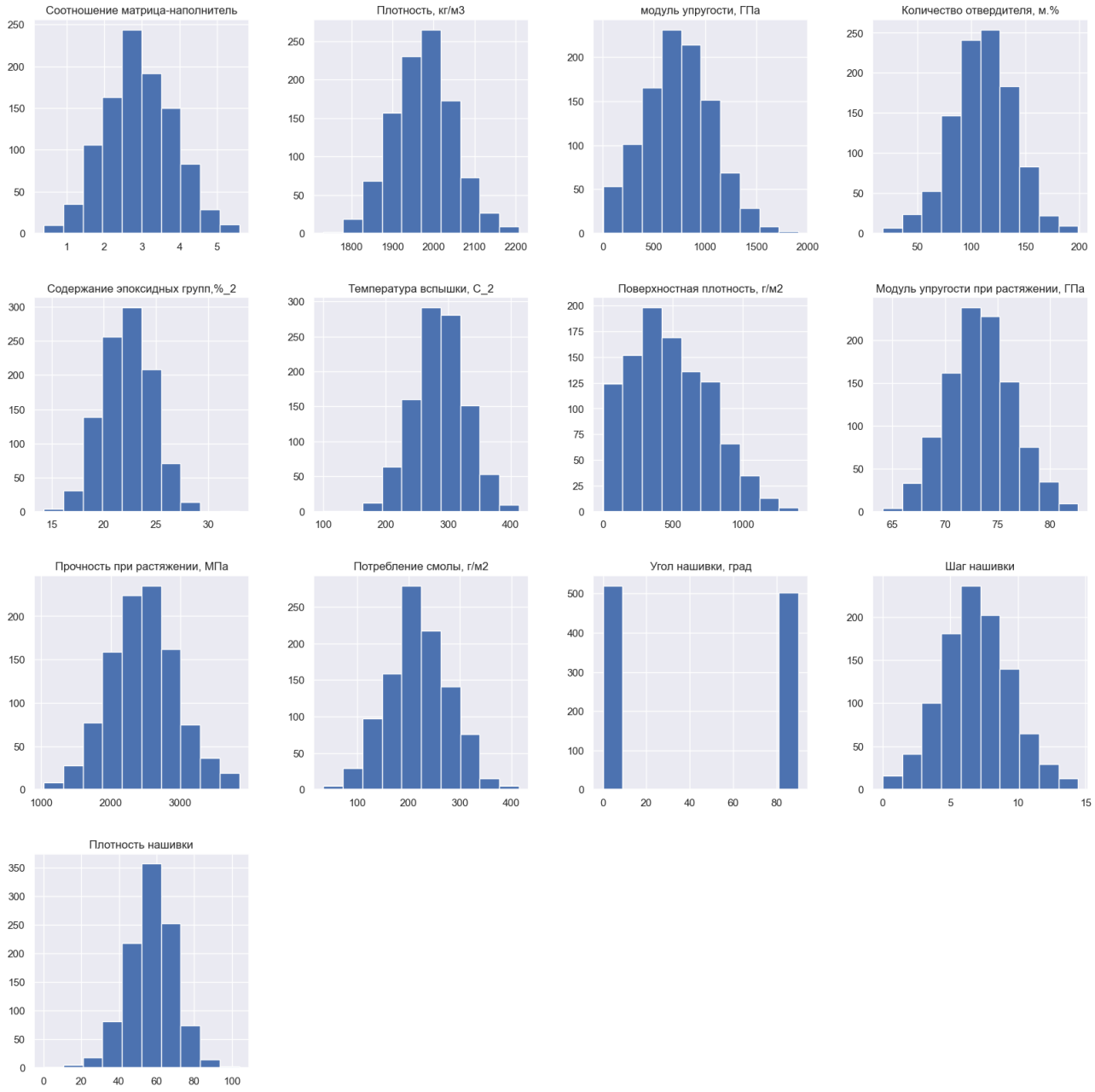

Рисунок 2 – Гистограммы распределения признаков

За исключением двух величин, величины датасета подчиняются нормальному распределению, как видно из рисунка 2. Гистограмма для поверхностной плотности также близка к нормальному распределению, но демонстрирует смещение вправо. Гистограмма для угла нашивки демонстрирует дискретный характер распределения.

При помощи библиотеки Seaborn построены диаграммы размаха (ящики с усами) для каждой величины (Seaborn.boxplot()). Диаграммы приведены на рисунке 3.

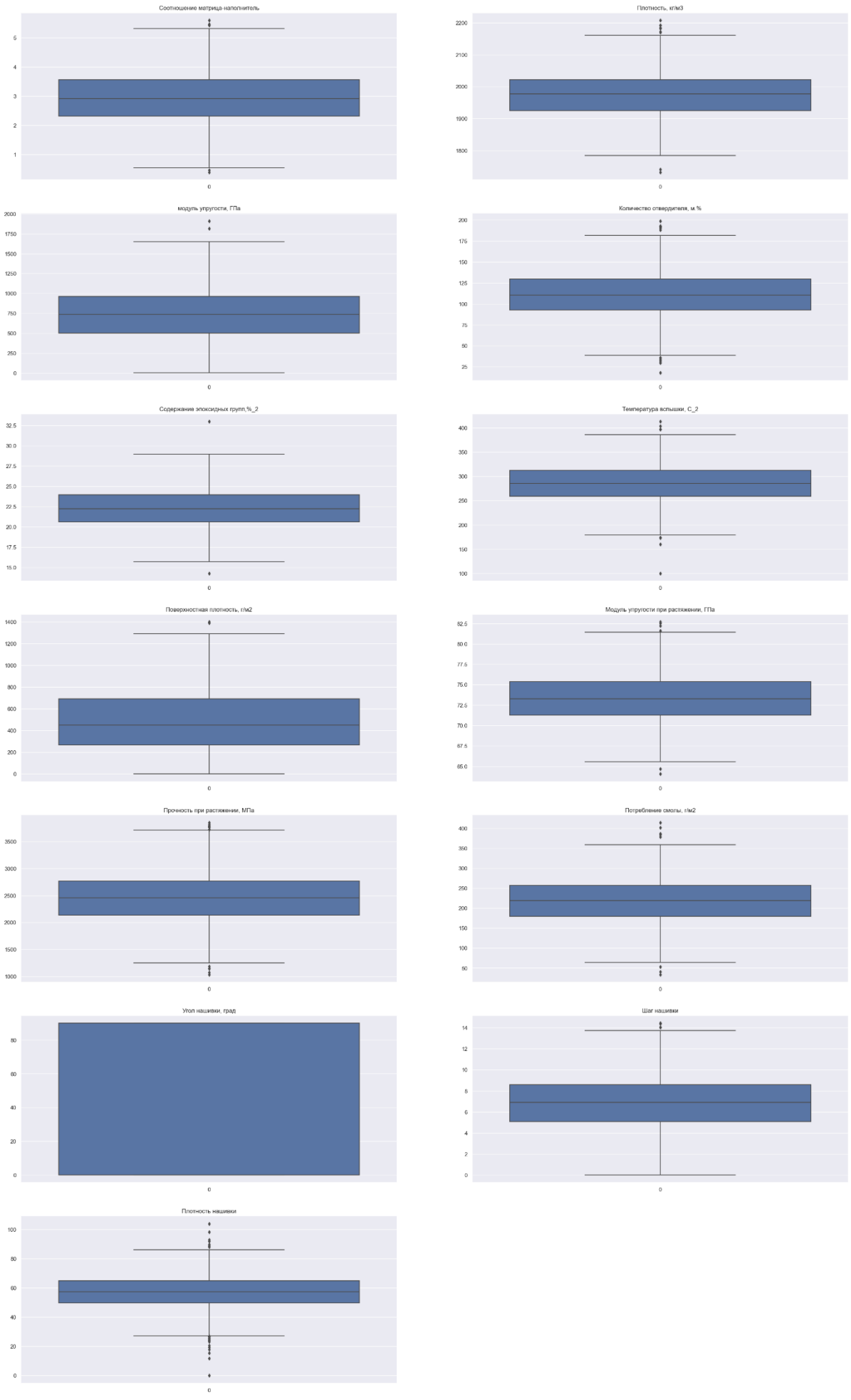

Рисунок 3 – Диаграммы размаха

Диаграмма размаха наглядно показывает медиану, нижний и верхний квартили, минимальное и максимальное значение выборки. Расстояния между различными частями ящика позволяют визуально оценить степень разброса (дисперсии) и асимметрии данных и выявить выбросы.

Выбросы наблюдаются по всем параметрам, кроме угла нашивки. Так как угол нашивки принимает дискретные значения, то диаграмма размаха для него не показательна.

Оценка взаимной корреляции в датасете проведена при помощи построения парных графиков рассеяния величин (Seaborn.pairplot) и тепловой карты (Seaborn.heatmap). Приведены на рисунках 4 и 5 соответственно.

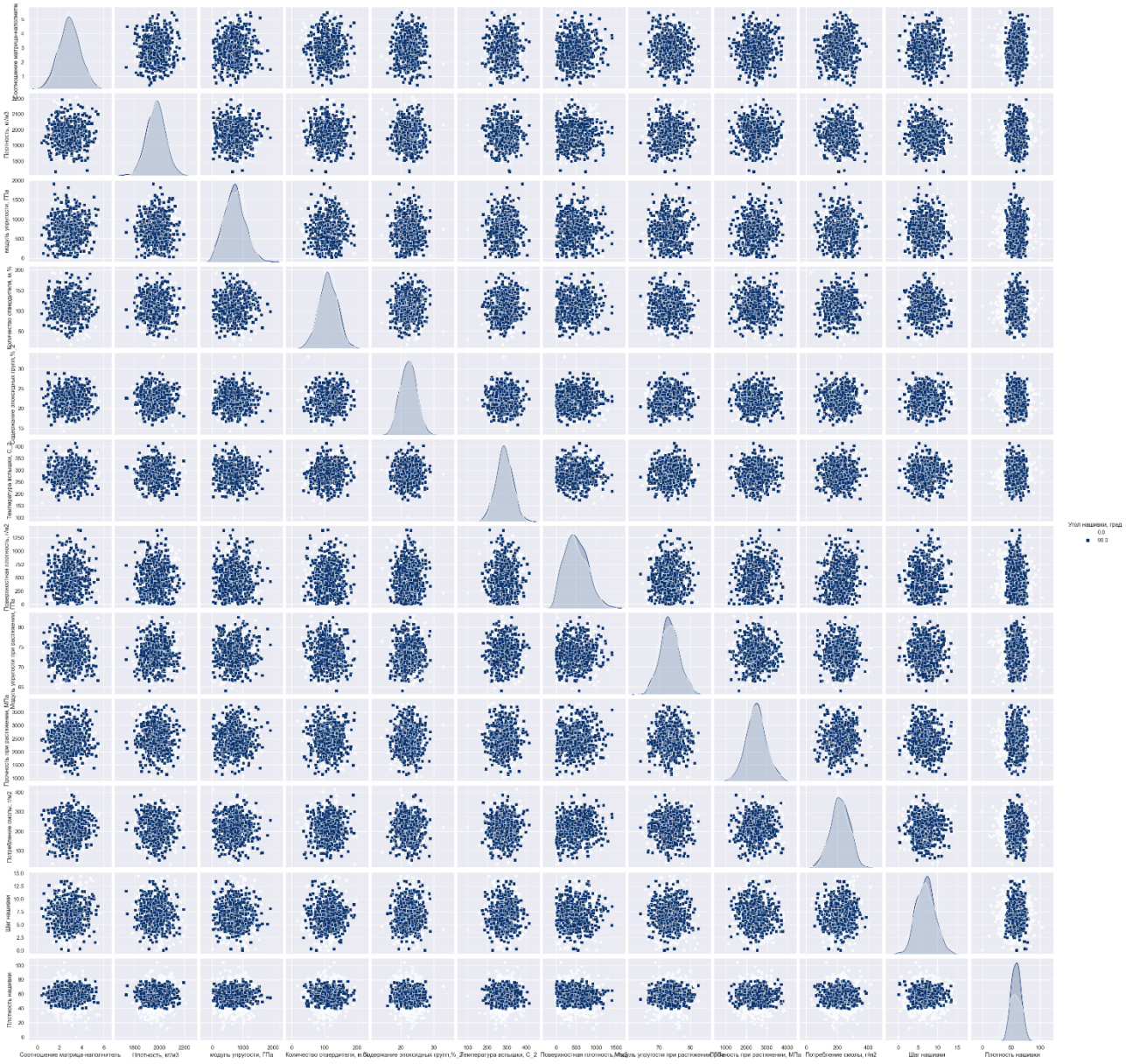

Рисунок 4 – парные графики рассеяния величин

Из рисунка 4 можно сделать вывод об отсутствии значительной взаимной корреляции между данными. О числовом выражении корреляции можно судить по тепловой карте (Рисунок 5)

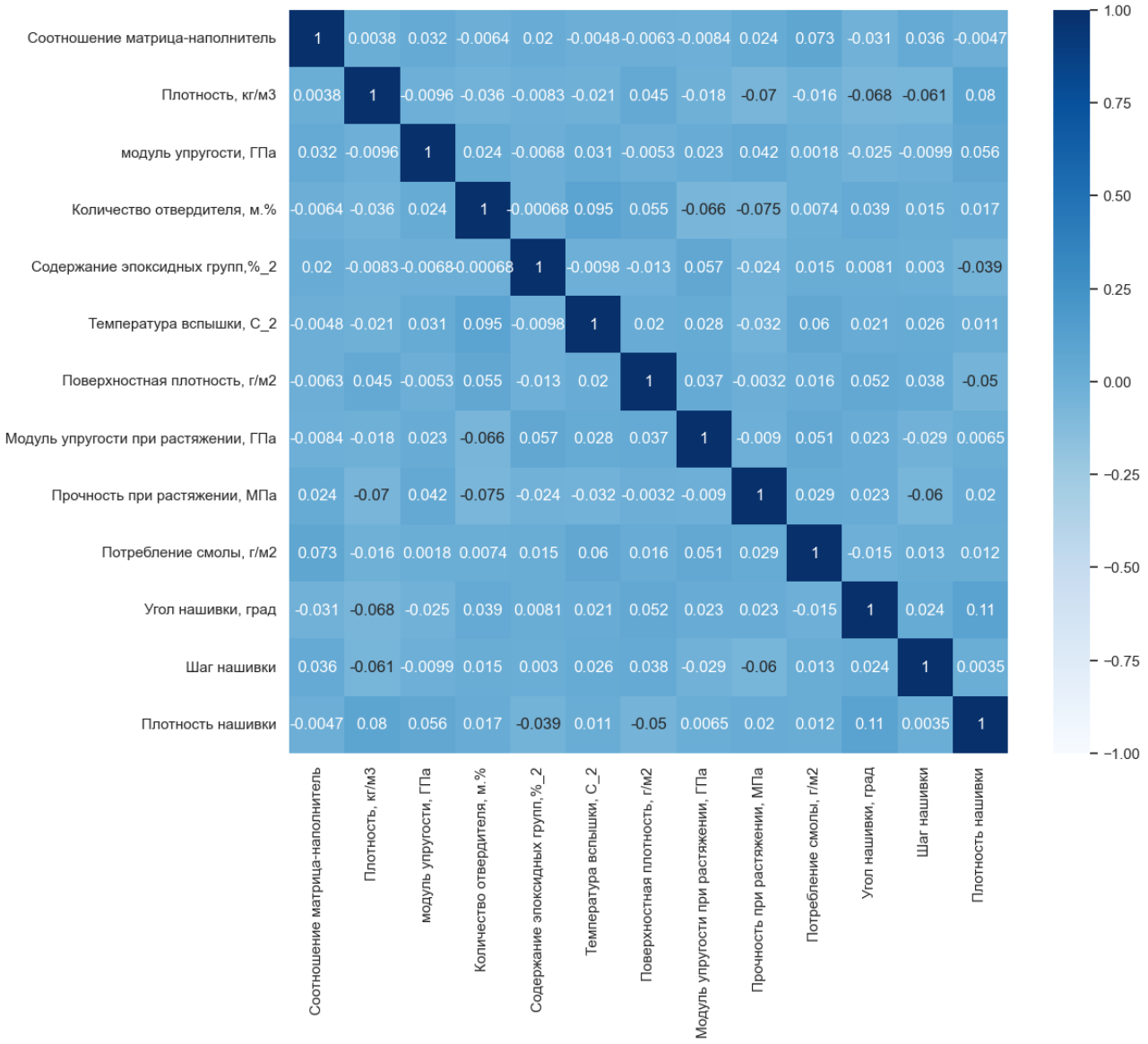

Рисунок 5 – тепловая карта коэффициентов корреляции

Из тепловой карты коэффициентов корреляции видно, что значения коэффициентов корреляции не превышают 0.2, из чего можно сделать вывод, что переменные в датасете являются независимыми. Преимущественно коэффициенты корреляции меньше 0.1, за исключением плотности нашивки и угла нашивки, составляющем 0.107, что также не говорит о наличии значительной корреляции между значениями.

Разведочный анализ показал наличие выбросов. В основном переменные подчиняются распределению близкому к нормальному. Переменные датасета можно считать независимыми.

2 Практическая часть

2.1 Предобработка данных

На основании результатов разведочного анализа данных проведена предобработка датасета. Подсчитано количество выбросов для каждого признака. Полученные значение приведены в таблице 1.

Таблица 1 – количество выбросов в данных

Признак	Первая итерация	Вторая итерация	Третья итерация
Соотношение матрица-наполнитель	6	0	0
Плотность, кг/м3	9	0	0
модуль упругости, ГПа	2	1	0
Количество отвердителя, м.%	14	0	0
Содержание эпоксидных групп,%_2	2	0	0
Температура вспышки, С_2	8	0	0
Поверхностная плотность, г/м2	2	0	0
Модуль упругости при растяжении, ГПа	6	1	0
Прочность при растяжении, МПа	11	4	2
Потребление смолы, г/м2	8	1	1
Угол нашивки, град	0	0	0
Шаг нашивки	4	0	0
Плотность нашивки	21	3	1

Количество выбросов незначительно относительно общего количества данных, поэтому их можно удалить из датасета. После удаления строк, содержащих выбросы, число строк в датасете составляет 922.

Для дальнейшей разработки и обучения модели была выполнена нормализация данных с помощью MinMaxScaler библиотеки Scikit learn. Для нормализованных данных повторно приведены некоторые статистические характеристики.

На рисунке приведена описательная статистика нормализованного датасета, по которой можно убедиться в успешной нормализации данных.

	count	mean	std	min	25%	50%	75%	max
Соотношение матрица-наполнитель	936.0	0.498933	0.187489	0.0	0.372274	0.494538	0.629204	1.0
Плотность, кг/м3	936.0	0.502695	0.187779	0.0	0.368517	0.511229	0.624999	1.0
модуль упругости, ГПа	936.0	0.446764	0.199583	0.0	0.301243	0.447061	0.580446	1.0
Количество отвердителя, м.%	936.0	0.504664	0.188865	0.0	0.376190	0.506040	0.637978	1.0
Содержание эпоксидных групп,%_2	936.0	0.491216	0.180620	0.0	0.367716	0.489382	0.623410	1.0
Температура вспышки, С_2	936.0	0.516059	0.190624	0.0	0.386128	0.515980	0.646450	1.0
Поверхностная плотность, г/м2	936.0	0.373733	0.217078	0.0	0.205619	0.354161	0.538683	1.0
Модуль упругости при растяжении, ГПа	936.0	0.488647	0.191466	0.0	0.359024	0.485754	0.615077	1.0
Прочность при растяжении, МПа	936.0	0.495706	0.188915	0.0	0.365149	0.491825	0.612874	1.0
Потребление смолы, г/м2	936.0	0.521141	0.195781	0.0	0.392067	0.523766	0.652447	1.0
Угол нашивки, град	936.0	0.511752	0.500129	0.0	0.000000	1.000000	1.000000	1.0
Шаг нашивки	936.0	0.502232	0.183258	0.0	0.372211	0.504258	0.624604	1.0
Плотность нашивки	936.0	0.513776	0.191342	0.0	0.390482	0.516029	0.638842	1.0

Рисунок 6 - описательная статистика нормализованного датасета

Для нормализованного датасета повторно построены:

– Гистограммы распределения признаков (рисунок 7)
– Диаграммы размаха (рисунок 8)

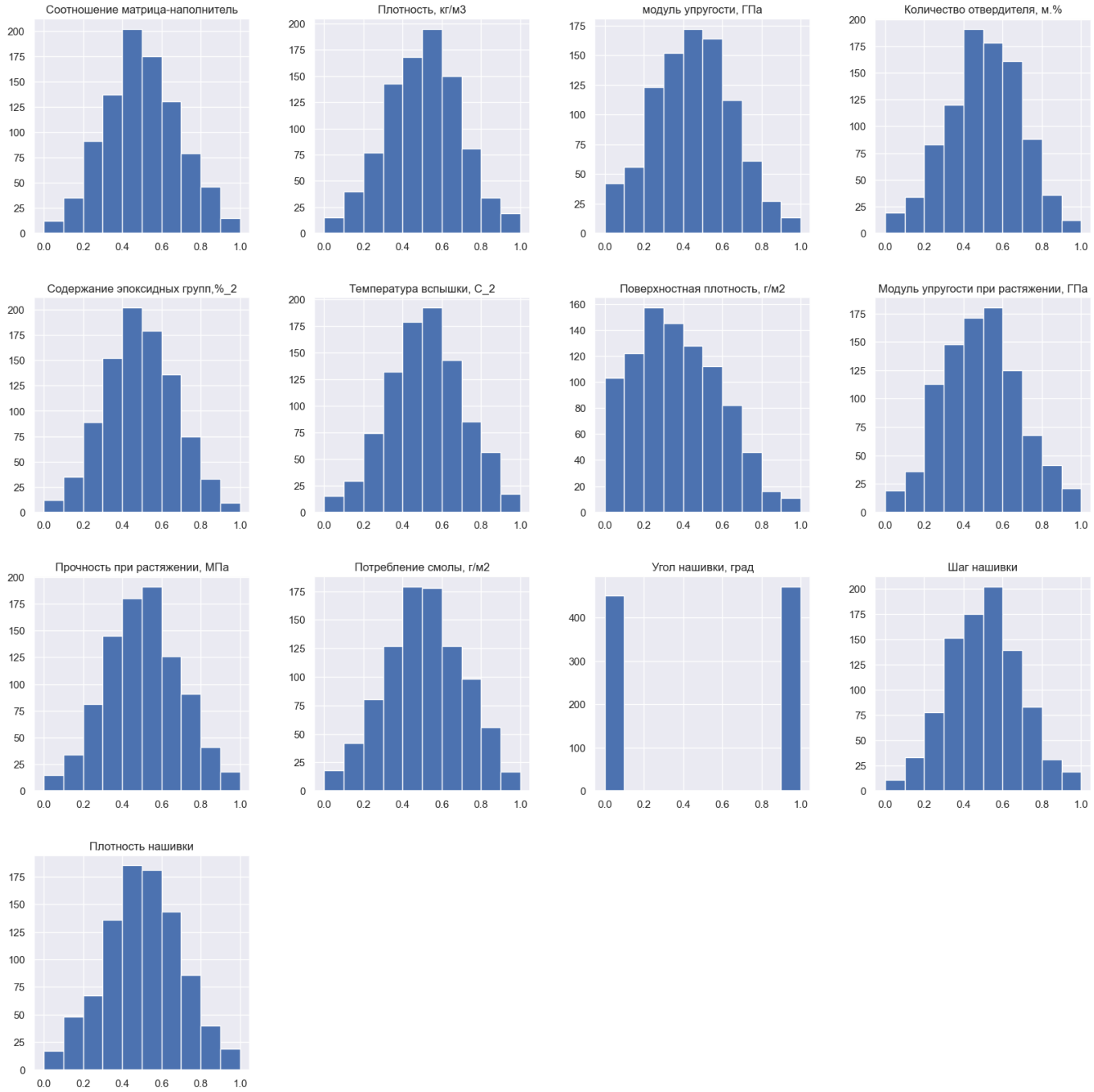

Рисунок 7. Гистограммы распределения признаков

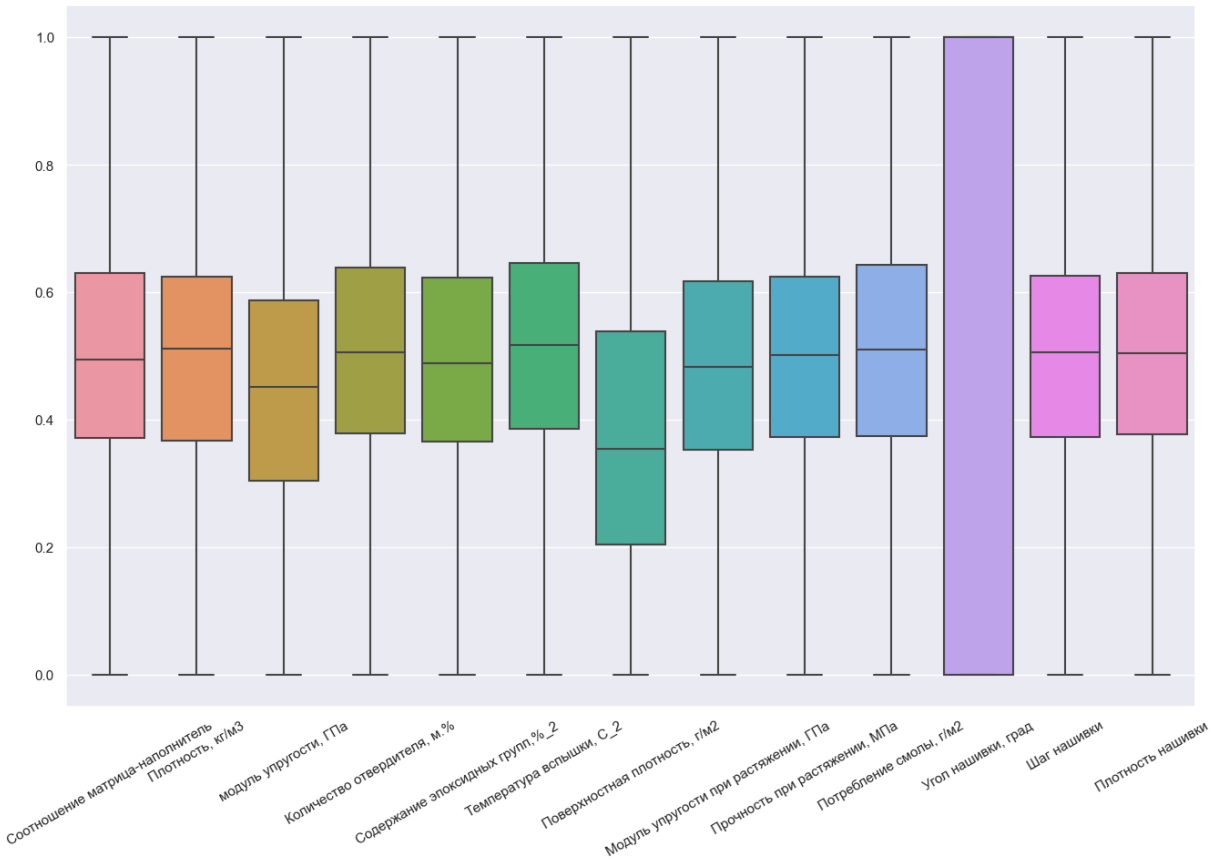

Рисунок 8. Диаграммы размаха

Диаграммы размаха приведены на одном рисунке. В нормализованном датасете эти диаграммы позволяют визуально оценить разброс признаков и убедиться в отсутствии выбросов.

2.2 Разработка и обучение модели

Обучение моделей производится в библиотеке машинного обучения Scikit learn. Обучены модели регрессии для модуля упругости при растяжении и прочности при растяжении. Для всех моделей, кроме линейной регрессии осуществлен поиск параметров по сетке с перекрестной проверкой с помощью Scikit learn GridSearchCV.

Разбивка датасета на тренировочную и тестовую выборки произведена при помощи Scikit learn train_test_split() в соотношении 70:30.

Размер обучающей выборки: 645.

Размер тестовой выборки: 277.

Модели Scikit learn, использованные в работе приведены в таблице 2.

Таблица 2 – использованные модели машинного обучения

Модель	Реализация Scikit learn
линейная регрессия	linear_model.LinearRegression
случайный лес	ensemble.RandomForestRegressor
метод k-ближайших соседей	Neighbors.KNeighborsRegressor()
метод опорных векторов	svm.SVR()
метод градиентного бустинга	ensemble.GradientBoostingRegressor

2.3 Тестирование модели

Оценка моделей производится по средней абсолютной ошибке (MAE) и по средней квадратичной ошибке (MSE).

Применяются встроенные метрики Scikit learn.

Сравнение моделей приведены в таблицах 3 и 4.

Таблица 3 – сравнение моделей регрессии модуля упругости при растяжении

Модель	MAE	MSE
линейная регрессия	0.161937	0.039564
случайный лес	0.161887	0.039612
метод k-ближайших сосе дей	0.161696	0.039254
метод опорных векторов	0.160573	0.039016
метод градиентного буст инга	0.161973	0.039844

Таблица 4 – сравнение моделей регрессии прочности при растяжении

Модель	MAE	MSE
линейная регрессия	0.149937	0.034523
случайный лес	0.150998	0.034852
метод k-ближайших сосе дей	0.152226	0.035300
метод опорных векторов	0.151112	0.034843
метод градиентного буст инга	0.152338	0.035206

Из приведенных сравнений видно, что ни один из примененных методов не дал результата, значительно отличающегося от остальных. В таком случае можно сделать выбор в пользу наиболее простого и быстрого метода – линейной регрессии. Также возможно продолжить эксперимент с настройкой параметров методов.

2.4 Нейронная сеть

Нейронная сеть для рекомендации соотношения «матрица-наполнитель» разрабатывается с использованием библиотеки TensorFlow и фреймворка Keras.

Используется нейронная сеть прямого распространения (feedforward reural network - FNN), т.е. классическая модель с несколькими полно связными слоями. Результат работы слоя становится входом следующего и т.д.

Гиперпараметры модели приведены на рисунке 9.

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 normalization (Normalizati  (None, 12)                3
 on)

 dense (Dense)               (None, 128)               1664

 dense_1 (Dense)             (None, 128)               16512

 dense_2 (Dense)             (None, 128)               16512

 dense_3 (Dense)             (None, 64)                8256

 dense_4 (Dense)             (None, 64)                4160

 dense_5 (Dense)             (None, 32)                2080

 dense_6 (Dense)             (None, 16)                528

 dense_7 (Dense)             (None, 1)                 17

=================================================================
Total params: 49732 (194.27 KB)
Trainable params: 49729 (194.25 KB)
Non-trainable params: 3 (16.00 Byte)
_________________________________________________________________
```

Рисунок 9. Архитектура модели

Обучение модели происходило за 150 эпох, размер батча - 32. Параметры подобраны вручную. На рисунке 10 приведен результат изменения MSE нейронной сети во время обучения. MSE уменьшается со временем по мере выполнения алгоритма, но затем выходит на плато, не изменяясь значительно.

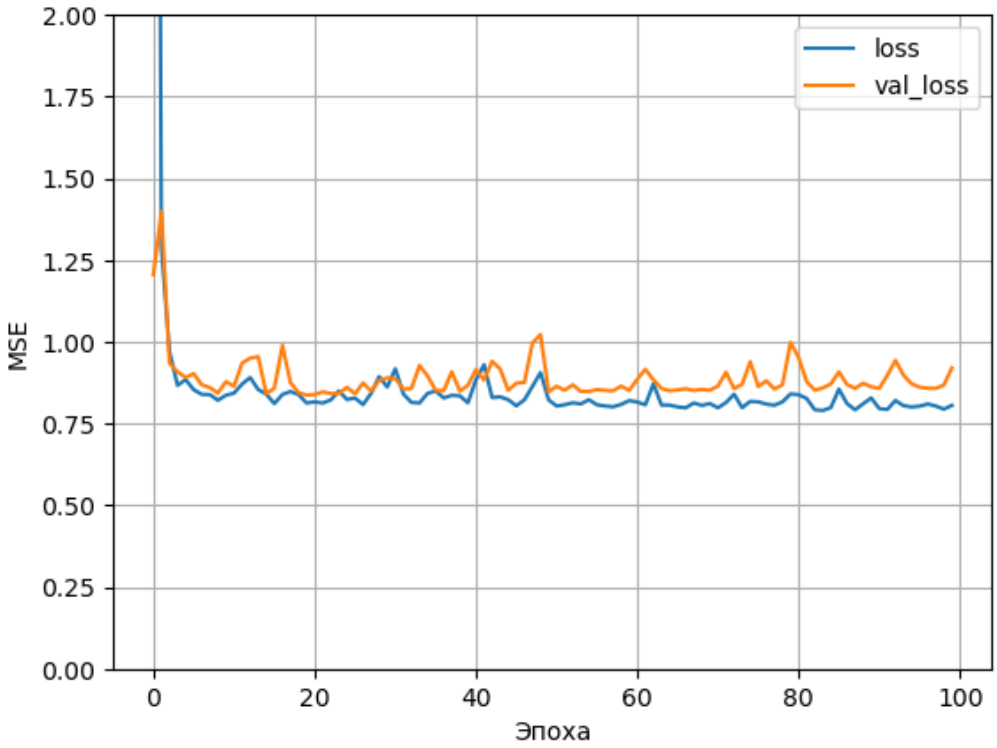

Рисунок 10 – MSE во время обучения нейросети

Увеличение числа эпох приводит к переобучению модели, т.е. уменьшению ошибки при обучении, но увеличению ошибки на валидационных данных.

Гистограмма распределения ошибки приведена на рисунке 11.

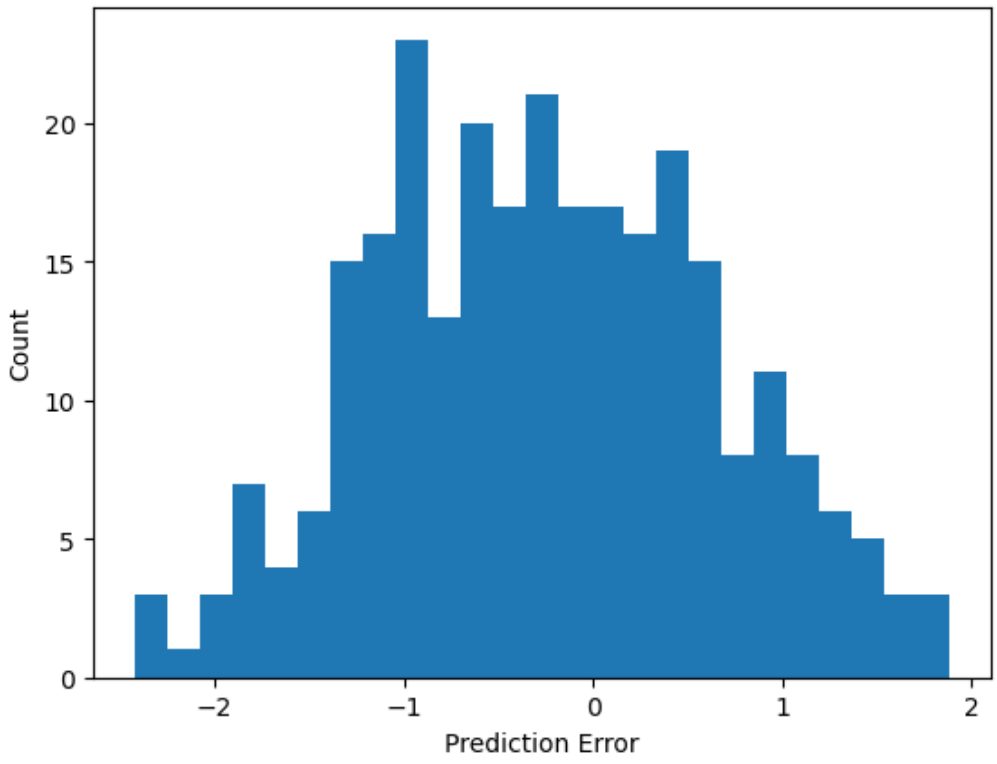

Рисунок 10 – Распределение ошибки

Из гистограммы на рисунке 10 видно, что ошибка распределена в основном в интервале [-1:1], при этом наибольшее в распределении приходится не на ноль. Из этого можно сделать вывод, что модель не является оптимальной для решения этой задачи.

При уменьшении числа нейронов или слоев модель стремится предсказывать среднее значение прогнозируемого признака во всех случаях.

Также, независимо от количества скрытых слоев и числа нейронов в них, модели FNN стремятся дать результатом среднее значение прогнозируемого признака. На рисунке 11 приведено для наглядности распределение прогнозов одной и промежуточных моделей. Как видно, прогнозные значения образуют прямую, проходящую в районе среднего значения величины соотношения «матрица-наполнитель». Аналогичное распределение для финальной модели приведено на рисунке 12.

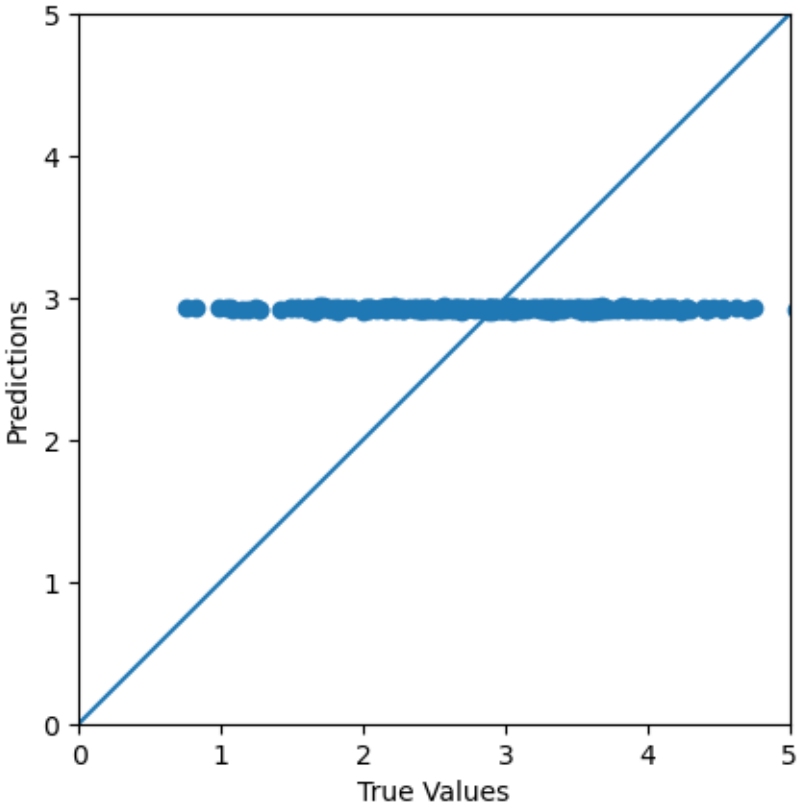

Рисунок 11. Предсказанные значения промежуточной модели

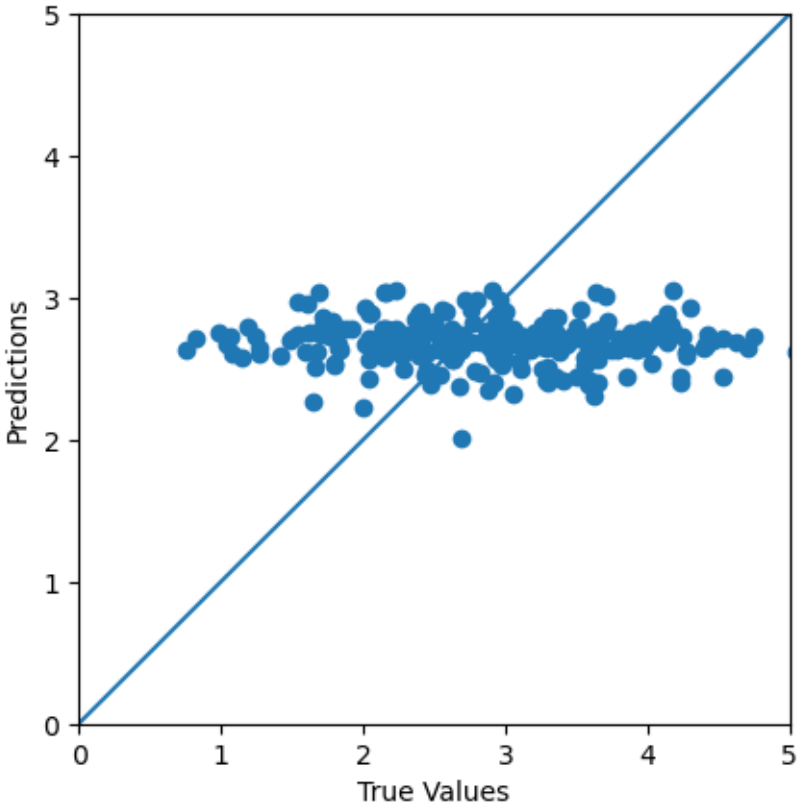

Рисунок 12. Предсказанные значения финальной модели

Модель не смогла обучиться с достаточной точностью. Среднее абсолютное отклонение (MAE) для модели составляет 0.7640, среднее абсолютное процентное отклонение (MAPE) - 0,30.

Можно сделать вывод о том, что регрессионная FNN модель не подходит для решения данной задачи. Тем более стоит отметить, что нейронные сети больше подходят для задач классификации.

2.5 Разработка приложения

Приложение было разработано с помощью Flask.

В интерфейсе использованы наименования параметров из датасета (скриншот приведен на рисунке 13).

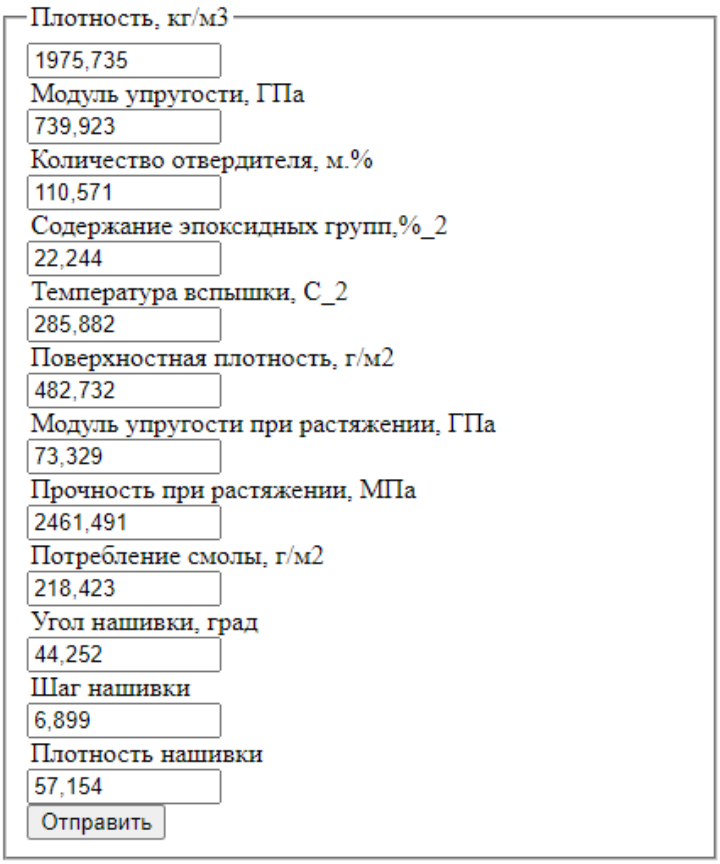

Рисунок 13 – Интерфейс приложения

Поля ввода по умолчанию заполнены средним значением признака в датасете.

При введении данных и после нажатия кнопки «отправить» приложение на основании обученной нейронной сети выводит рекомендуемое значение соотношения «матрица – наполнитель».

3 Создание удаленного репозитория

Созданный репозиторий на GitHub:
https://github.com/vinogradov96ma/data-science-PRO

Заключение

Разведочный анализ исходного датасета, а также результаты обучения регрессионных моделей показали, что датасет является предобработанным, данные в нём не коррелируют между собой и по сути датасет и не содержит реальных значений для отработки обучения и тренировки моделей.

Полученная модель нейронной показывает низкие результаты и позволяет предсказывать только значения, близкие к средним значениям прогнозируемого параметра.

Библиографический список

1. Л.И. Бондалетова, В.Г. Бондалетов Полимерные композиционные материалы: - Режим доступа - https://portal.tpu.ru/SHARED/b/BONDLI/stud_work/p_k_m_m/Tab1/Posobie_PCM.pdf. (дата обращения - 05.09.2023).
2. 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. (дата обращения: 05.09.2023)
3. Andre Ye. 5 алгоритмов регрессии в машинном обучении, о которых вам сле-дует знать: – Режим доступа: https://habr.com/ru/company/vk/blog/513842/(дата обращения: 06.09.2023).
4. Документация по библиотеке keras: – Режим доступа: https://keras.io/api/.(дата обращения: 05.09.2023).
5. Документация по библиотеке matplotlib: – Режим доступа: https://matplotlib.org/stable/users/index.html. (дата обращения: 10.09.2023)
6. Документация по библиотеке numpy: – Режим доступа: https://numpy.org/doc/1.22/user/index.html#user. (дата обращения: 05.09.2023).
7. Документация по библиотеке pandas: – Режим доступа: https://pandas.pydata.org/docs/user_guide/index.html#user-guide. (дата обращения: 05.09.2023).
8. Документация по библиотеке scikit-learn: – Режим доступа: https://scikit-learn.org/stable/user_guide.html. (дата обращения 09.09.2023).
9. Документация по библиотеке seaborn: – Режим доступа: https://seaborn.pydata.org/tutorial.html. (дата обращения 08.09.2023).
10. Документация по библиотеке Tensorflow: – Режим доступа: https://www.tensorflow.org/overview (дата обращения: 08.09.2023).
11. Документация по языку программирования python: – Режим доступа: https://docs.python.org/3.8/index.html. (дата обращения: 05.09.2023).
12. Краткий обзор алгоритма машинного обучения Метод Опорных Векторов (SVM) – Режим доступа: https://habr.com/ru/post/428503/ (дата обращения 05.09.2023)
13. Плас Дж. Вандер, Python для сложных задач: наука о данных и машинное обучение. Санкт-Петербург: Питер, 2018, 576 с.
14. Руководство по быстрому старту в flask: – Режим доступа: https://flask-russian-docs.readthedocs.io/ru/latest/quickstart.html. (дата обращения: 13.09.2022)