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»

Тема: «Прогнозирование конечных свойств новых материалов (композиционных материалов)»

Слушатель Желдак Евгений Михайлович

Москва, 2022

Содержание

Введение

Композиционные материалы представляют собой сочетание нескольких химически разнородных компонентов с границей раздела между ними. За счет выбора комбинаций армирующих компонентов и наполнителей, изменения их объемных долей, размеров, формы, ориентации и прочности связи по границе раздела результирующие свойства композита могут меняться в значительных пределах. Возможность получения материалов с уникальными свойствами не присущими отдельным компонентам обуславливает широкое применение композиционных материалов в различных областях техники.

Традиционно разработка композитных материалов – это долгосрочный процесс, так как по характеристикам отдельных компонентов невозможно рассчитать итоговые свойства композита. Для получения заданных свойств требуется большое количество испытаний различных комбинаций, что делает актуальной задачу прогнозирования успешное решение которой позволило бы снизить расходы и трудозатраты по разработке новых материалов.

Сформулируем цель работы: создание системы прогнозирования параметров композитного материала на основе методов машинного обучения. Для достижения поставленной цели необходимо решить следующие задачи:

1) изучить теоретические основы и методы машинного обучения (регрессии);
2) провести разведочный анализ экспериментальных данных;
3) выполнить подготовку обучающих и тестовых выборок (предобработку данных);
4) сформировать и обучить несколько моделей для прогноза модуля упругости при растяжении и прочности при растяжении;
5) разработать нейронную сеть, для рекомендации (прогноза) соотношения матрица-наполнитель;
6) разработать приложение с графическим интерфейсом или интерфейсом командной строки, для выполнения прогноза соотношения матрица-

наполнитель по произвольно введённым пользователем входным параметрам;

7) оценить точность модели на тренировочном и тестовом наборах данных.

1. Аналитическая часть

1.1. Постановка задачи

Необходимо сделать модели для прогноза модуля упругости при растяжении, прочности при растяжении и соотношения матрица-наполнитель по входным параметрам из файлов с данными о параметрах базальтопластика X_bp.xlsx и нашивки углепластика X_nup.xlsx. По заданию файлы были объединены по индексу с типом объединения INNER. Результирующий набор данных содержит 13 признаков и 1023 строки (Таблица 1). Пропусков в данных нет. Все признаки, кроме «Угол нашивки», являются непрерывными, количественными, имеют вещественный тип. «Угол нашивки» принимает только два значения и будет закодирован как категориальный признак.

Таблица 1 – Список признаков

Признак	Число уникальных значений
Соотношение матрица-наполнитель	1014
Плотность, кг/м3	1013
модуль упругости, Гпа	1020
Количество отвердителя, м.%	1005
Содержание эпоксидных групп,%_2	1004
Температура вспышки, С_2	1003
Поверхностная плотность, г/м2	1004
Модуль упругости при растяжении, Гпа	1004
Прочность при растяжении, Мпа	1004
Потребление смолы, г/м2	1003
Угол нашивки, град	2
Шаг нашивки	989
Плотность нашивки	988

Гистограммы распределения переменных и диаграммы «ящик с усами» приведены на рисунках 1-2. По ним видно, что все признаки, кроме «Угол нашивки», имеют распределение близкое к нормальному и принимают неотрицательные значения. «Угол нашивки» принимает значения: 0, 90. По диаграммам «Ящики с усами» видно наличие выбросов, более подробно их можно рассмотреть в нормализованных данных (Рисунок 3). Далее при предобработке данных выбросы будут удалены.

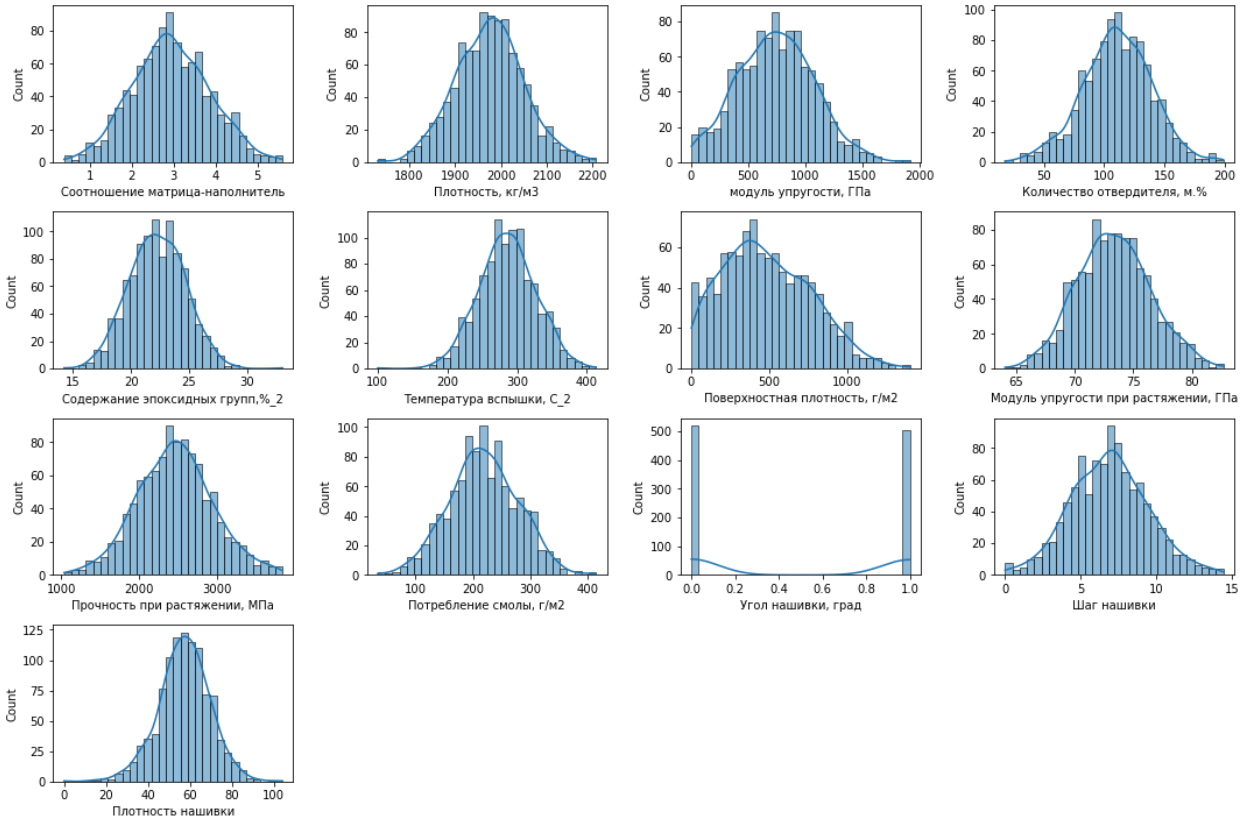

Рисунок 1 - Гистограммы распределения переменных

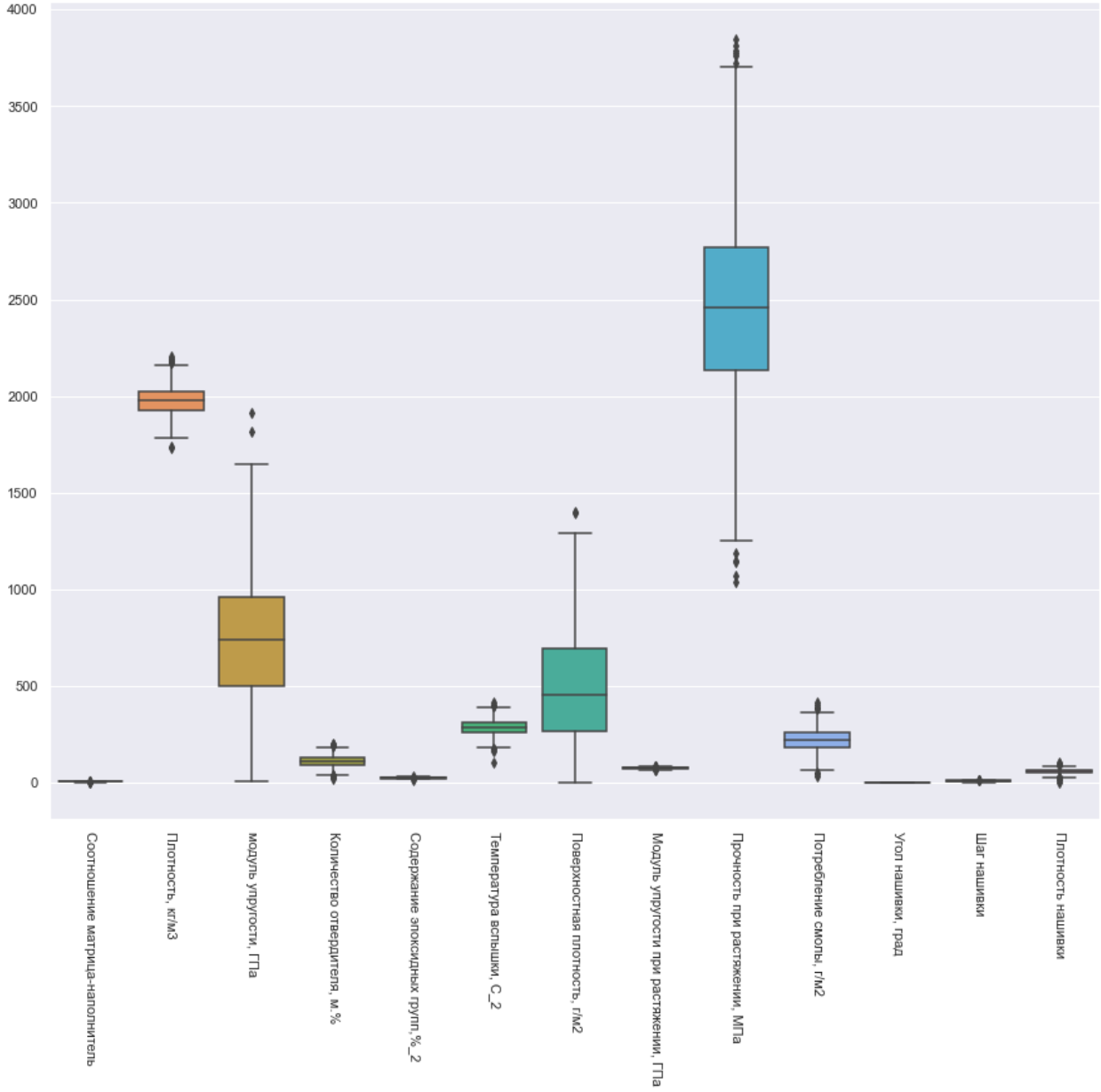

Рисунок 2 – «Ящики с усами»

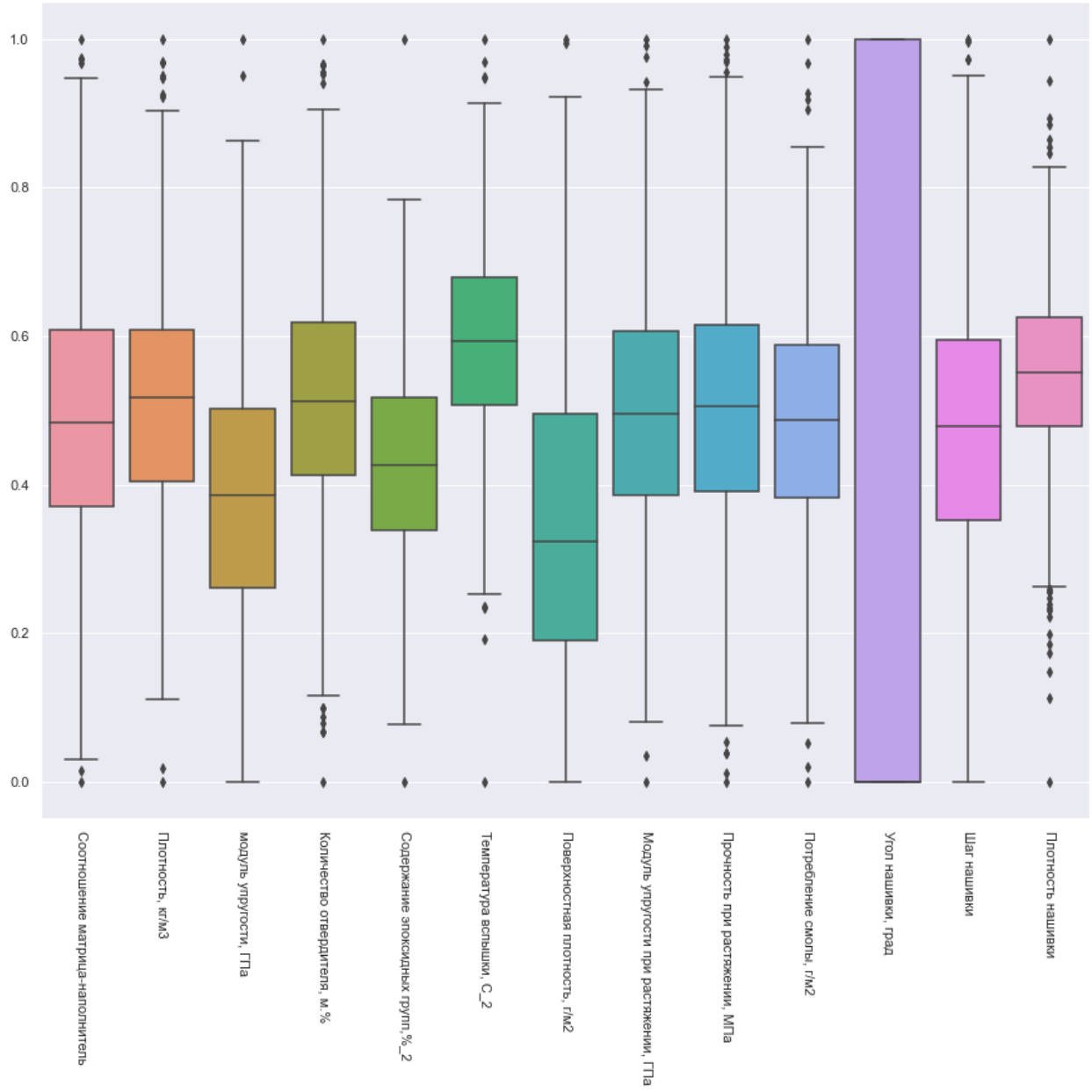

Рисунок 3 – «Ящики с усами» для нормализованных данных

Численные значения показателей описательной статистики (медиана, среднее, стандартное отклонение, минимум, максимум, квартили) представлены в Таблице 2.

Таблица 2 — Описательная статистика признаков датасета

Признак	count	mean	std	min	25%	50%	75%	max
Соотношение матрица-наполнитель	1023.0	2,93	0,91	0,39	2,32	2,91	3,55	5,59
Плотность, кг/м3	1023.0	1975,73	73,73	1731,76	1924,16	1977,62	2021,37	2207,77
модуль упругости, ГПа	1023.0	739,92	330,23	2,44	500,05	739,66	961,81	1911,54

Количество отвердителя, м.%	1023.0	110,57	28,30	17,74	92,44	110,56	129,73	198,95
Содержание эпоксидных групп,%_2	1023.0	22,24	2,41	14,25	20,61	22,23	23,96	33,00
Температура вспышки, С_2	1023.0	285,88	40,94	100,00	259,07	285,90	313,00	413,27
Поверхностная плотность, г/м2	1023.0	482,73	281,31	0,60	266,82	451,86	693,23	1399,54
Модуль упругости при растяжении, ГПа	1023.0	73,33	3,12	64,05	71,25	73,27	75,36	82,68
Прочность при растяжении, МПа	1023.0	2466,92	485,63	1036,86	2135,85	2459,52	2767,19	3848,44
Потребление смолы, г/м2	1023.0	218,42	59,74	33,80	179,63	219,20	257,48	414,59
Угол нашивки, град	1023.0	44,25	45,02	0,00	0,00	0,00	90,00	90,00

Попарные графики рассеяния точек приведены на рисунке 4. По ним также видны точки, отстоящие от основного облака данных – выбросы. Также по графикам можно сказать, что линейная зависимость между переменными выражена слабо.

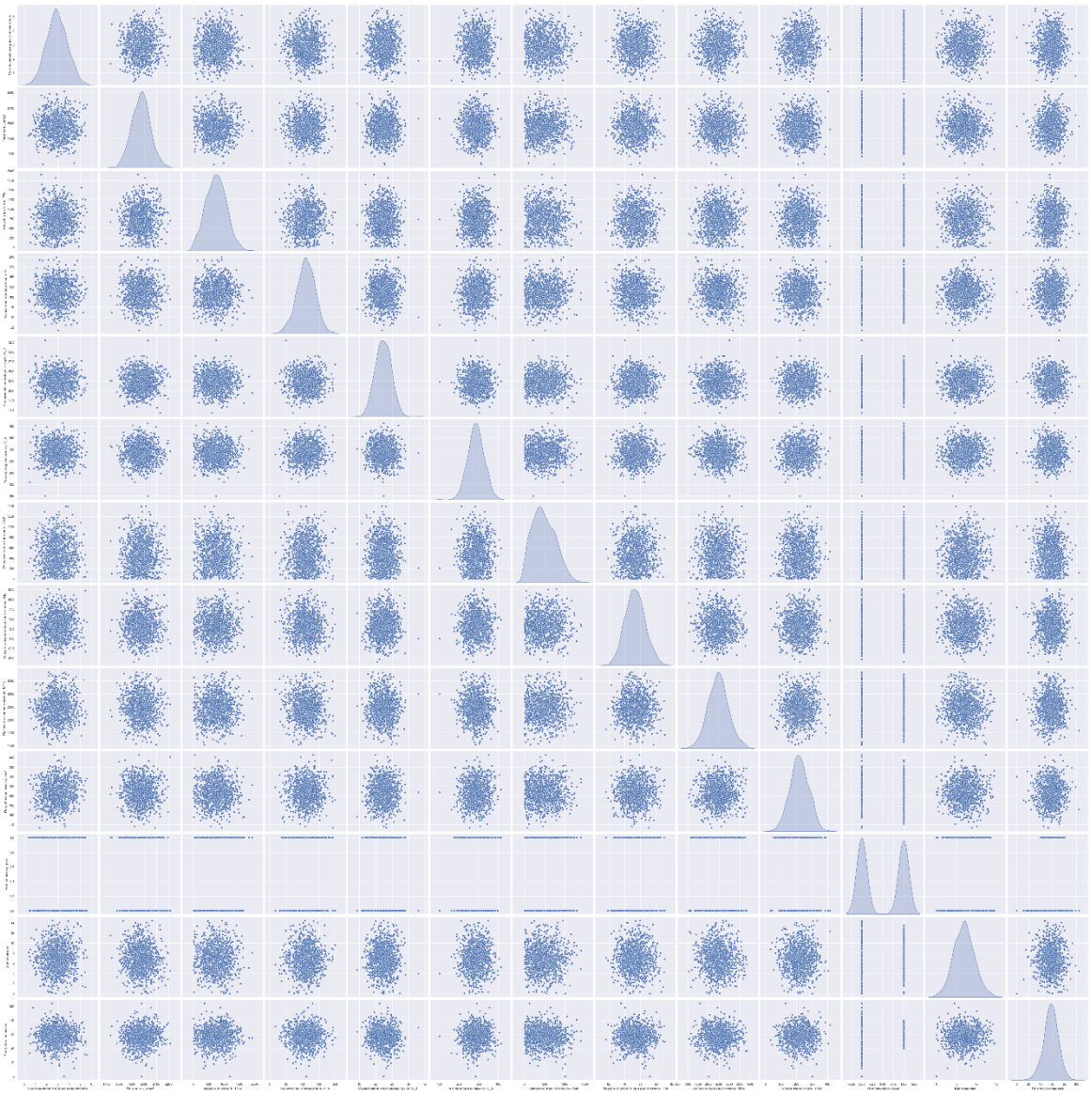

Рисунок 4 — Попарные диаграммы рассеяния точек

1.2. Описание используемых методов

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. Для построения моделей использовались библиотеки Scikit-learn, Keras и Tensorflow.

1.2.1 Линейная регрессия

За базовую модель для прогнозирования всех искомых параметров принята sklearn.linear_model.LinearRegression. Обычная линейная регрессия методом наименьших квадратов. LinearRegression соответствует линейной модели с коэффициентами w = (w1, …, wp), чтобы минимизировать остаточную сумму квадратов между наблюдаемыми целями в наборе данных и целями, предсказанными линейным приближением.

Простая линейная регрессия имеет место, если рассматривается зависимость между одной входной и одной выходной переменными. Для этого определяется уравнение регрессии и строится соответствующая прямая, известная как линия регрессии $y = ax + b$.

Коэффициенты a и b, называемые также параметрами модели, определяются таким образом, чтобы сумма квадратов отклонений точек, соответствующих реальным наблюдениям данных, от линии регрессии была бы минимальной. Коэффициенты обычно оцениваются методом наименьших квадратов.

Если ищется зависимость между несколькими входными и одной выходной переменными, то имеет место множественная линейная регрессия. Соответствующее уравнение имеет вид:

$Y = b0 + b1 * x1 + b2 * x2 + \cdots + bn * xn,$

где n - число входных переменных.

Очевидно, что в данном случае модель будет описываться не прямой, а гиперплоскостью. Коэффициенты уравнения множественной линейной регрессии подбираются так, чтобы минимизировать сумму квадратов отклонения реальных точек данных от этой гиперплоскости.

1.2.2 Полиномиальная регрессия

Простая линейная регрессия может быть расширена путем построения полиномиальных функций из коэффициентов. В случае стандартной линейной

регрессии у вас может быть модель, которая выглядит следующим образом для двумерных данных:

$$\hat{y}(w, x) = w_0 + w_1 x_1 + w_2 x_2$$

Если мы хотим подогнать к данным параболоид, а не плоскость, мы можем объединить функции в полиномы второго порядка, чтобы модель выглядела так:

$$\hat{y}(w, x) = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_1 x_2 + w_4 x_1^2 + w_5 x_2^2$$

Наблюдение заключается в том, что это все еще линейная модель : чтобы убедиться в этом, представьте, что вы создаете новый набор функций.

$$z = [x_1, x_2, x_1 x_2, x_1^2, x_2^2]$$

С этой перемаркировкой данных наша проблема может быть записана

$$\hat{y}(w, z) = w_0 + w_1 z_1 + w_2 z_2 + w_3 z_3 + w_4 z_4 + w_5 z_5$$

Мы видим, что полученная полиномиальная регрессия относится к тому же классу линейных моделей, который мы рассмотрели выше (т.е. модель линейна now) и могут быть решены теми же методами. Рассматривая линейные соответствия в многомерном пространстве, построенном с помощью этих базовых функций, модель обладает гибкостью, позволяющей соответствовать гораздо более широкому диапазону данных.

1.2.3 Случайный лес

Случайный лес (RandomForest) — представитель ансамблевых методов, комбинирующий множество отдельных деревьев решений. Формула итогового решателя — это усреднение предсказаний отдельных деревьев.

$$a(x) = \frac{1}{N} \sum_{i=1}^{N} b_i(x)$$

,

где

N – количество деревьев;

i – счетчик для деревьев;

b – решающее дерево;

x – сгенерированная нами на основе данных выборка.

Для определения входных данных каждому дереву используется метод случайных подпространств. Базовые алгоритмы обучаются на различных подмножествах признаков, которые выделяются случайным образом.

Преимущества случайного леса:

– высокая точность предсказания;

– редко переобучается;

– практически не чувствителен к выбросам в данных;

– одинаково хорошо обрабатывает как непрерывные, так и дискретные признаки, данные с большим числом признаков;

– высокая параллелизуемость и масштабируемость.

Из недостатков можно отметить, что его построение занимает больше времени. Так же теряется интерпретируемость.

1.2.4 Многослойный перцептрон

Нейронная сеть — это последовательность нейронов, соединенных между собой связями. Нейронные сети применяются для решения задач регрессии, классификации, распознавания образов и речи, компьютерного зрения и других. На настоящий момент это самый мощный, гибкий и широко применяемый инструмент в машинном обучении. Структура нейронной сети пришла в мир программирования из биологии. Вычислительная единица нейронной сети — нейрон или персептрон.

У каждого нейрона есть определённое количество входов, куда поступают сигналы, которые суммируются с учётом значимости (веса) каждого входа. Смещение – это дополнительный вход для нейрона, который всегда равен 1 и, следовательно, имеет собственный вес соединения. Так же у нейрона есть функция активации, которая определяет выходное значение нейрона. Она

используется для того, чтобы ввести нелинейность в нейронную сеть. Примеры активационных функций: relu, сигмоида.

У полносвязной нейросети выход каждого нейрона подается на вход всем нейронам следующего слоя. У нейросети имеется:

– входной слой — его размер соответствует входным параметрам;

– скрытые слои — их количество и размерность определяем специалист;

– выходной слой — его размер соответствует выходным параметрам.

Прямое распространение – это процесс передачи входных значений в нейронную сеть и получения выходных данных, которые называются прогнозируемым значением. Прогнозируемое значение сравниваем с фактическим с помощью функции потери. В методе обратного распространения ошибки градиенты (производные значений ошибок) вычисляются по значениям весов в направлении, обратном прямому распространению сигналов. Значение градиента вычитают из значения веса, чтобы уменьшить значение ошибки. Таким образом происходит процесс обучения. Обновляются веса каждого соединения, чтобы функция потерь минимизировалась. Для обновления весов в модели используются различные оптимизаторы. Количество эпох показывает, сколько раз выполнялся проход для всех примеров обучения.

1.2.5 Метрики качества моделей

Существует множество различных метрик качества, применимых для регрессии. В этой работе используются:

– R2 или коэффициент детерминации измеряет долю дисперсии, объясненную моделью, в общей дисперсии целевой переменной. Если он близок к единице, то модель хорошо объясняет данные, если же он близок к нулю, то прогнозы сопоставимы по качеству с константным предсказанием;

– RMSE (Root Mean Squared Error) или корень из средней квадратичной ошибки принимает значениях в тех же единицах, что и целевая переменная.

Метрика использует возведение в квадрат, поэтому хорошо обнаруживает грубые ошибки, но сильно чувствительна к выбросам.

RMSE принимает положительные значения, её надо минимизировать. R2 в норме принимает положительные значения. Эту метрику надо максимизировать. Отрицательные значение коэффициента детерминации означают плохую объясняющую способность модели.

1.3. Разведочный анализ данных

Цель разведочного анализа данных — выявить закономерности в данных. Для корректной работы большинства моделей желательна сильная зависимость выходных переменных от входных и отсутствие зависимости между входными переменными. Для проведения разведочного анализа использованы библиотеки Numpy, Pandas, Matplotlib, Seaborn и Scikit-learn.

Проверку пропусков выполняли с метода df.info(), который показывает количество не нулевых значений и тип данных. Для визуализации распределения значений по каждому столбцу и взаимосвязи между данными использован sns.histplot, sns.boxplot, sns.pairplot. Метод df.nunique() возвращает количество уникальных значений для каждого столбца. Признак 'Угол нашивки, град' имеет всего 2 значения 0 и 90, которые закодированы с помощью LabelEncoder. Класс LabelEncoder используется для кодирования данные имеющих всего два варианта значений, одно из которых будет закодировано нулем, а второе единицей.

Отображение коэффициентов корреляции выполнено с помощью тепловой карты их значений sns.heatmap. Удаление выбросы производим, используя межквартильный размах IQR, по схеме

1) рассчитаем первый и третий квартиль (Q1 и Q3);
2) оценим межквартирный диапазон, IQR-Q1;
3) оценим нижнюю границу, значения меньше 1.5*IQR;
4) оценим верхнюю границу, значения больше 1.5*IQR;

5) заменим точки данных, которые лежат за пределами нижней и верхней границы на nan;
6) удалим пропуски nan, если их количество намного меньше количества значений.

Функция dropna() удаляет строки, при axis=0, или столбцы, при axis=1, с значениями NULL или NAN [7].

Для вывода диаграмм размаха («ящик с усами») в одних координатах применен MinMaxScale. Эстиматор MinMaxScaler преобразует значения признаков путем масштабирования в заданном диапазоне. Каждый признак масштабируется в отдельности таким образом, чтобы он находился в диапазоне между нулем и единицей.

2. Практическая часть

2.1. Предобработка данных

Цель препроцессинга, или предварительной обработки данных — обеспечить корректную работу моделей.

Категориальный признак один - 'Угол нашивки, град'. Он принимает значения 0 и 90. Модели отработают лучше, если мы превратим эти значения в 0 и 1 с помощью LabelEncoder или OrdinalEncoder.

Удаляем выбросы за пределами 1.5 IQR. Значения параметров описательной статистики после удаления выбросов показаны в Таблице 3. От исходных 1023 строк осталось 932 (что вполне достаточно для обучения моделей), выбросы составили примерно 10% от объёма выборки.

Таблица 3 – Описательная статистика после удаления выбросов

	count	mean	std	min	25%	50%	75%	max
Соотношение матрица-наполнитель	932.0	2,93	0,90	0,55	2,32	2,91	3,55	5,31
Плотность, кг/м3	932.0	1974,14	70,94	1784,48	1923,14	1977,45	2020,34	2161,57
модуль упругости, ГПа	932.0	738,31	329,20	2,44	498,58	738,74	961,65	1649,42
Количество отвердителя, м.%	932.0	110,97	26,97	38,67	92,52	111,11	130,00	181,83
Содержание эпоксидных групп,%_2	932.0	22,20	2,40	15,70	20,56	22,18	23,96	28,96

Температура вспышки, С_2	932.0	285,95	39,40	179,37	259,10	285,95	312,84	386,07
Поверхностная плотность, г/м2	932.0	482,86	279,66	0,60	266,98	457,73	694,90	1291,34
Модуль упругости при растяжении, ГПа	932.0	73,30	3,04	65,55	71,23	73,26	75,31	81,42
Прочность при растяжении, МПа	932.0	2464,10	459,22	1250,39	2146,94	2456,39	2752,35	3689,22
Потребление смолы, г/м2	932.0	217,70	57,79	63,69	179,49	218,25	256,75	359,05
Угол нашивки, град	932.0	0,51	0,50	0,00	0,00	1,00	1,00	1,00
Шаг нашивки	932.0	6,93	2,51	0,04	5,15	6,97	8,61	13,73
Плотность нашивки	932.0	57,47	11,20	27,89	50,22	57,58	64,80	86,01

На рисунке 5 и 6 показаны гистограммы и «ящики с усами» для очищенной выборки.

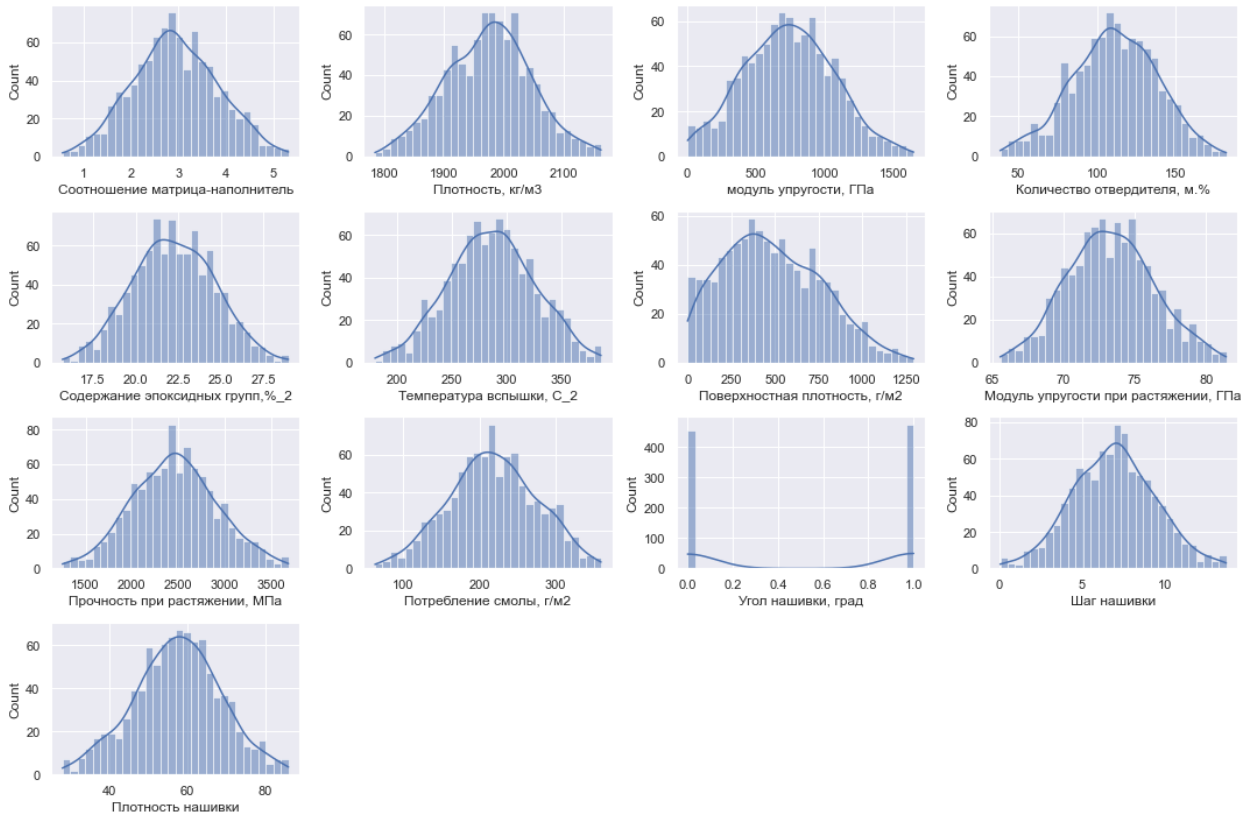

Рисунок 5 – Гистограммы после удаления выбросов

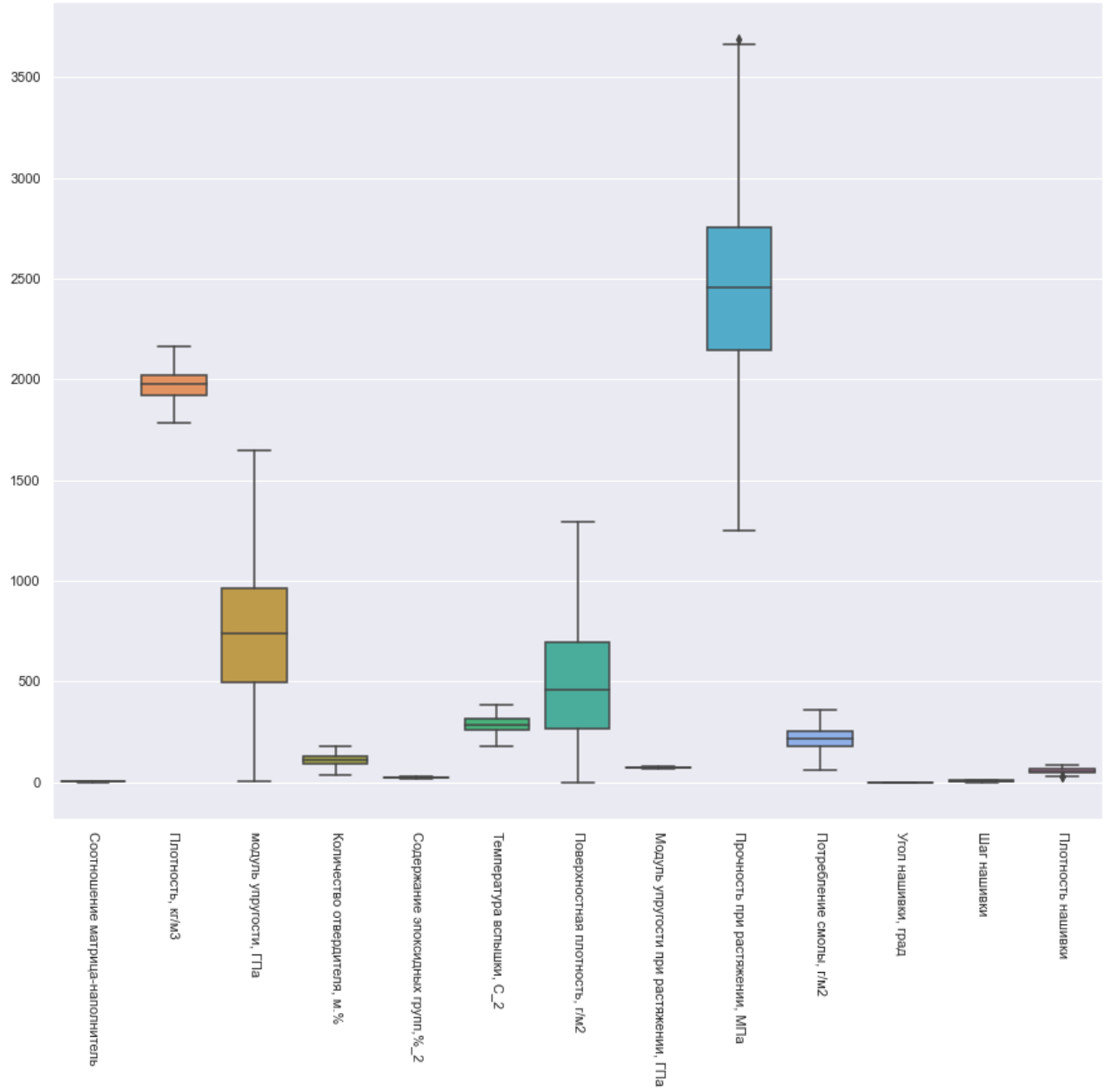

Рисунок 6 – «Ящики с усами» после удаления выбросов

Вещественных количественных признаков у нас большинство. Однако их значения лежат в разных диапазонах. Для сопоставимости данных выполним нормализацию – приведение в диапазон от 0 до 1 с помощью MinMaxScaler. Распределение нормализованных признаков показано на рисунке 7.

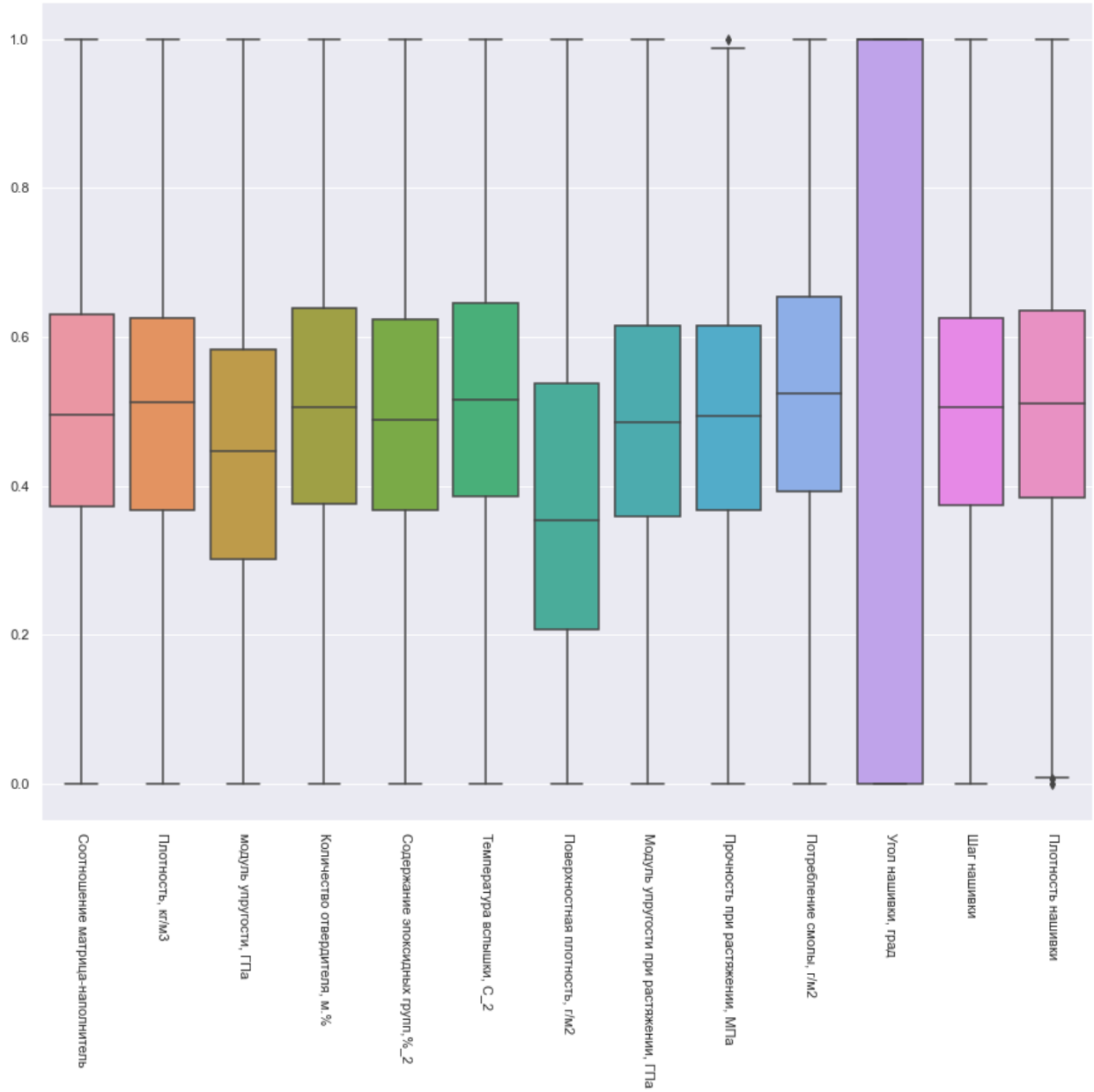

Рисунок 7 – «Ящики с усами» после нормализации

Построим попарные диаграммы рассеяния (Рисунок 8) и корреляционную матрицу (Рисунок 9). Так как данные близки к нормальным, используем для расчёта коэффициента используем метод Пирсона. По форме «облаков точек» каких-либо зависимостей между признаками не замечено. Согласно матрице корреляции коэффициенты корреляции близки к нулю, что означает отсутствие линейной зависимости между признаками. Можно предположить, что качество прогноза линейных моделей будет невысоким.

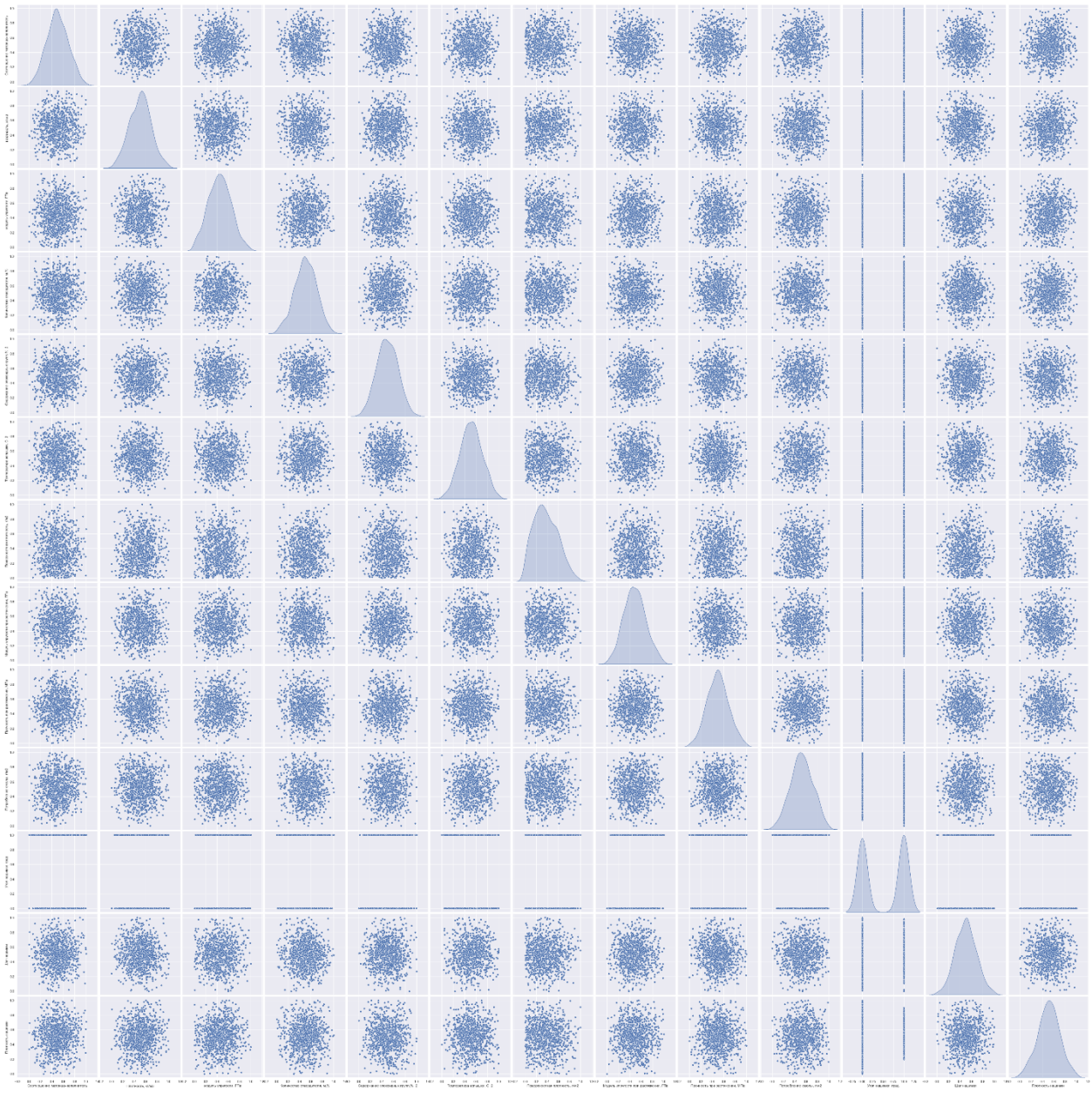

Рисунок 8 – Попарные диаграммы рассеяния после нормализации

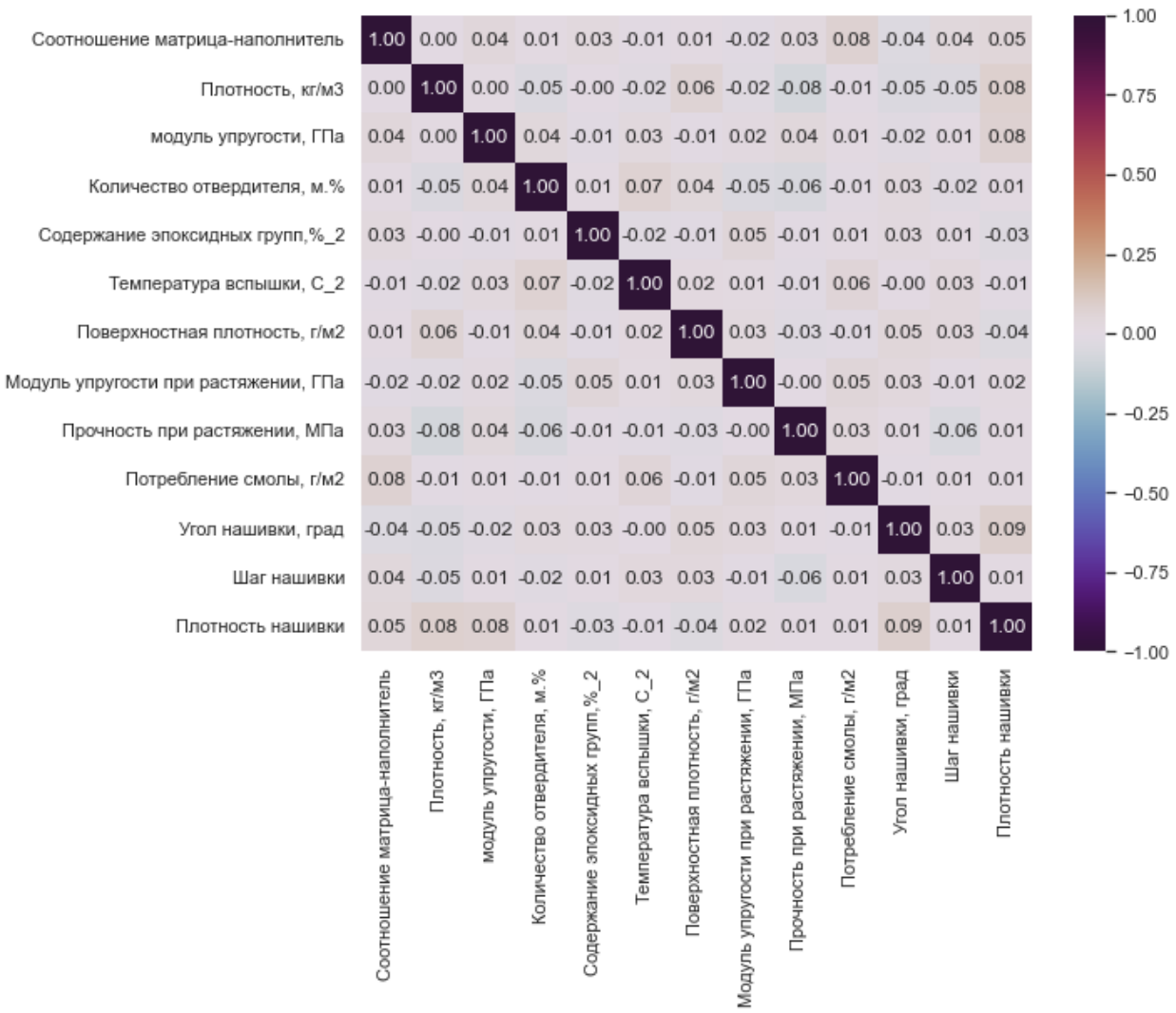

Рисунок 9 – Корреляционная матрица

2.2 Разработка и обучение регрессионных моделей

Для прогноза модуля упругости при растяжении и прочности при растяжении использованы модели LinearRegression, PolynomialRegression, RandomForestRegressor и MLPRegressor. Делим данные на тестовую и обучающую выборки согласно заданию. 30% данных оставим на тестирование модели, на остальных происходит обучение моделей. Зерно генератора случайных чисел зададим постоянным для воспроизводимости результатов обучения.

2.2.1 Прогнозирование модуля упругости при растяжении

Метрики моделей представлены в таблице 4. Ни одна из моделей не смогла удовлетворительно описать наши данные. Это может быть связано со сложной нелинейной зависимостью между переменными, либо с их зашумлённостью. Графики предсказанных (красным) и тестовых (синим) значений для различных моделей показаны на рисунках 10-13.

Таблица 4 – Метрики моделей регрессии

Модель	rmse	r2
Linear	0.185817	-0.021807
Polynomial	0.195041	-0.125768
Random Forest	0.187003	-0.034893
Multilayer Perceptron	0.204796	-0.241197

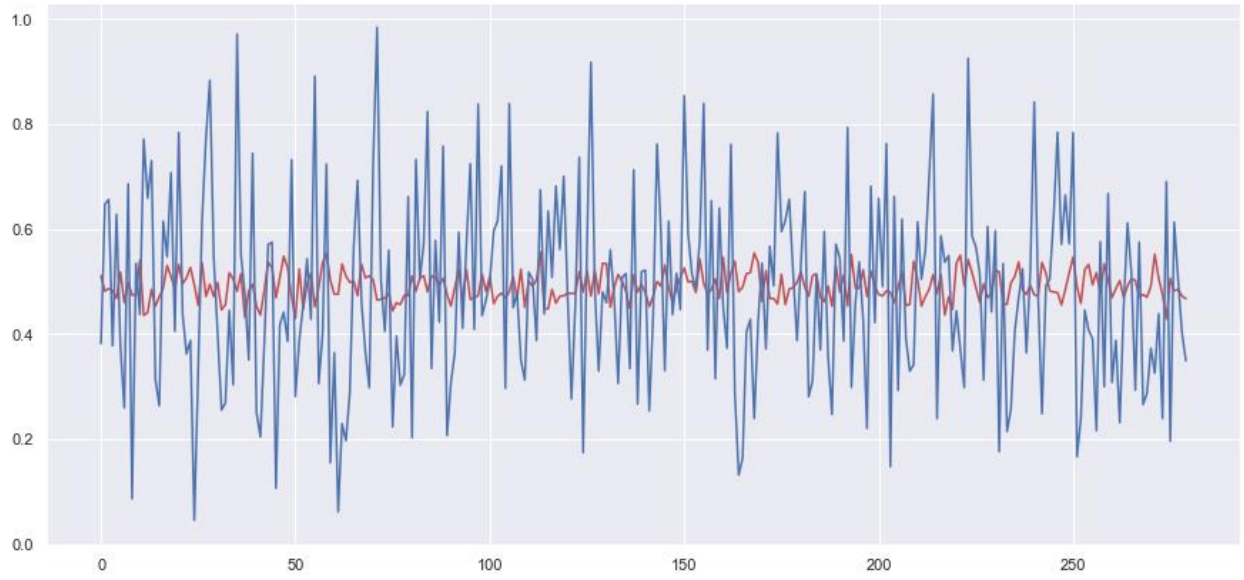

Рисунок 10 – Линейная регрессия

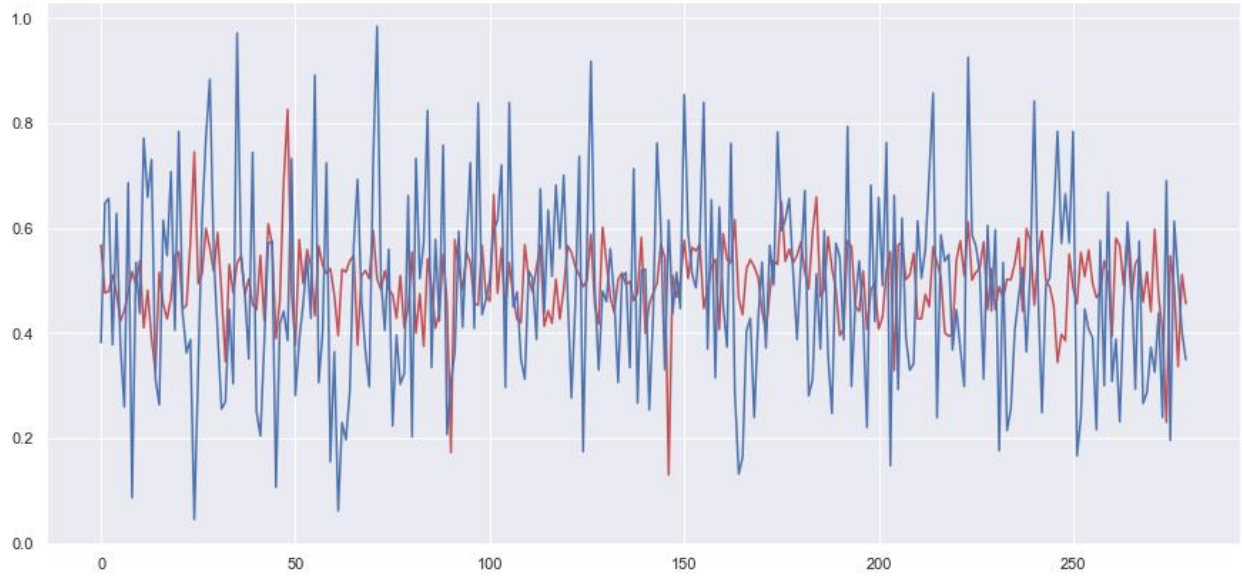

Рисунок 11 – Полиномиальная регрессия

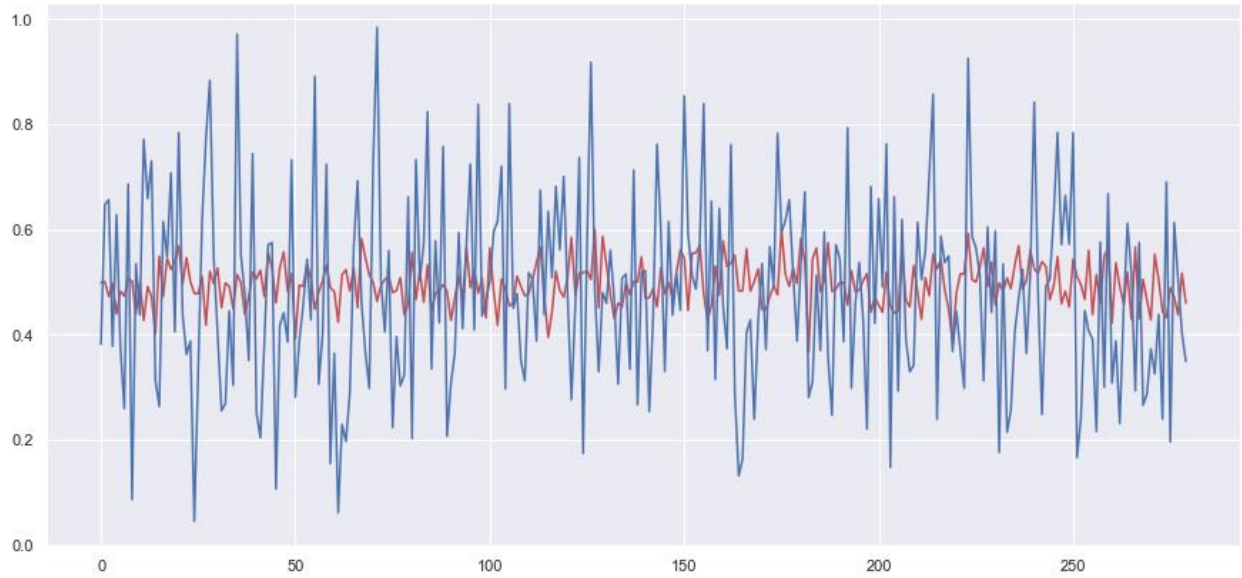

Рисунок 12 – Случайный лес

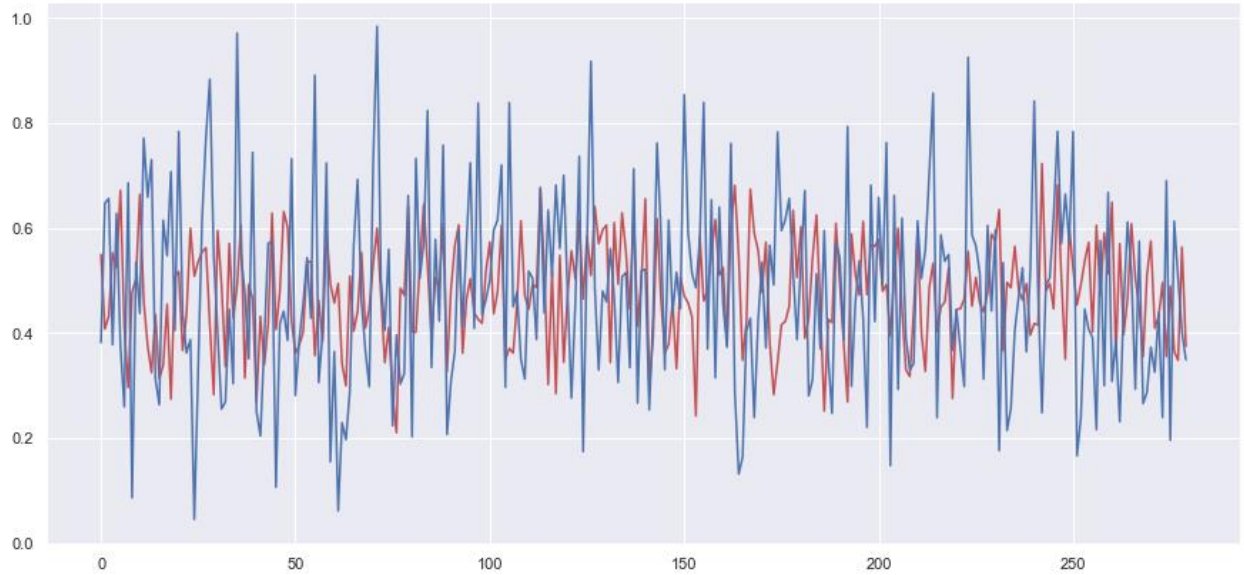

Рисунок 13 – Многослойный перцептрон

2.2.2 Прогнозирование прочности при растяжении

Модели прогноза прочности при растяжении показали такие же (неудовлетворительные) результаты, как и для модуля упругости при растяжении. Метрики моделей представлены в таблице 5. Графики предсказанных (красным) и тестовых (синим) значений для различных моделей показаны на рисунках 14-17.

Таблица 5 – Метрики моделей регрессии

Модель	rmse	r2
Linear	0.188691	0.004901
Polynomial	0.193815	-0.049882
Random Forest	0.186723	0.025550
Multilayer Perceptron	0.217275	-0.319415

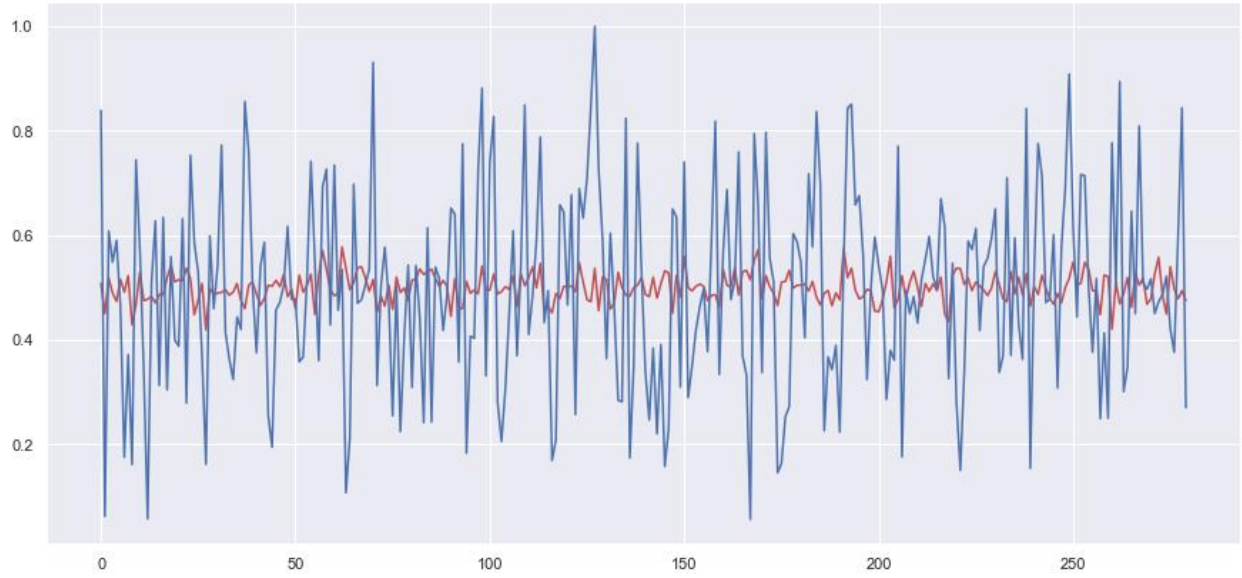

Рисунок 14 – Линейная регрессия

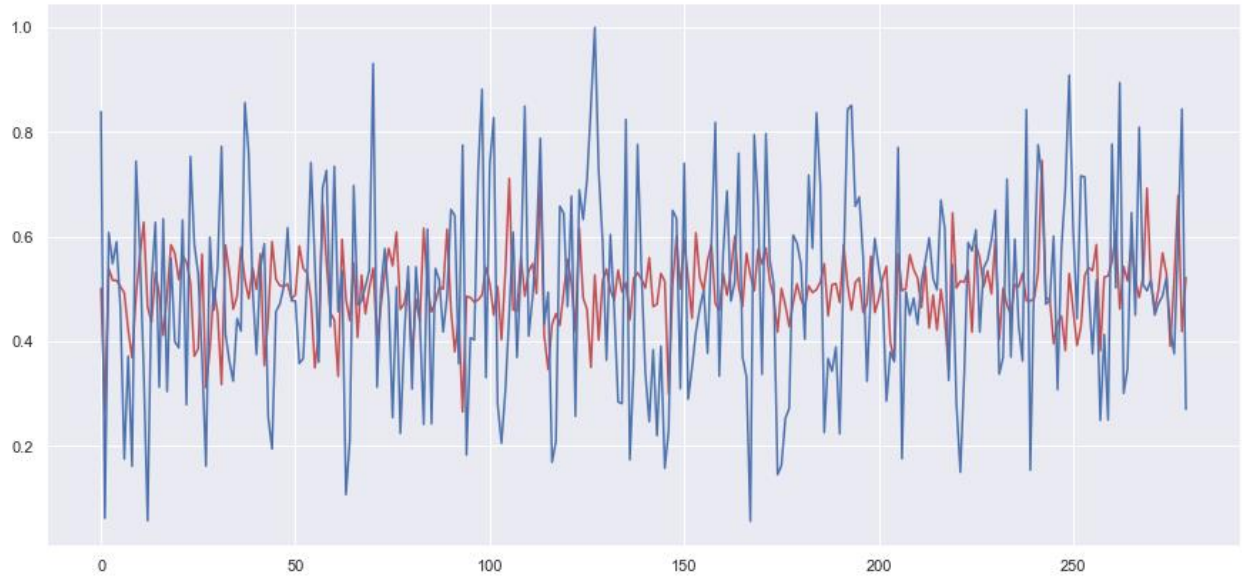

Рисунок 15 – Полиномиальная регрессия

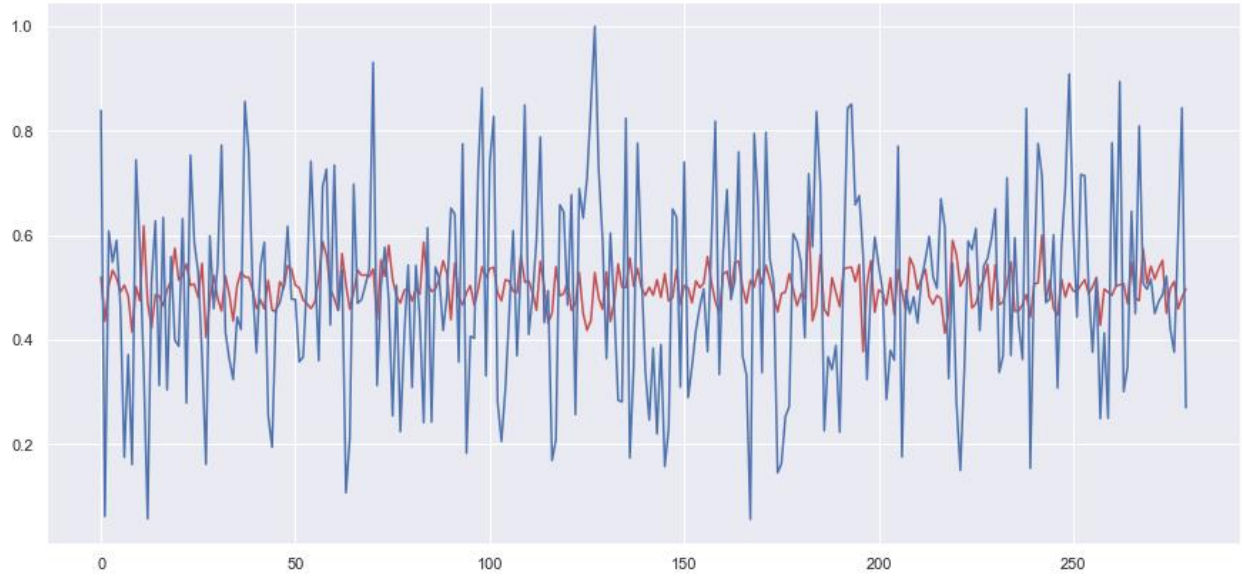

Рисунок 16 – Случайный лес

Рисунок 17 – Многослойный перцептрон

2.4 Разработка нейронной сети для прогнозирования соотношения матрица-наполнитель

По заданию для соотношения матрица-наполнитель необходимо построить нейросеть. Строю нейронную сеть с помощью класса keras.Sequential со следующими параметрами:

– входной слой нормализации 12 признаков;

– выходной слой для 1 признака;

– скрытых слоев: 1;

– нейронов в скрытом слое: 6;

– активационная функция скрытых слоев: relu;

– оптимизатор: Adam;

– loss-функция: MeanAbsoluteError.

Архитектура нейросети приведена на рисунке 18.

```
Model: "sequential_12"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 normalization_12 (Normaliza  (None, 12)               25
 tion)

 dense_29 (Dense)            (None, 6)                 78

 dense_30 (Dense)            (None, 1)                 7

=================================================================
Total params: 110
Trainable params: 85
Non-trainable params: 25
_________________________________________________________________
```

Рисунок 18 — Архитектура нейросети в виде summary

Запускаю обучение нейросети со следующими параметрами:

– пропорция разбиения данных на тестовые и валидационные: 30%;

– количество эпох: 100.

График ошибки обучения сети (RMSE) приведён на рисунке 19, сравнение прогноза с данными валидации на рисунке 20. Значения RMSE на тренировочной и проверочной выборках после обучения: 0.8232544660568237 и 1.0181834697723389.

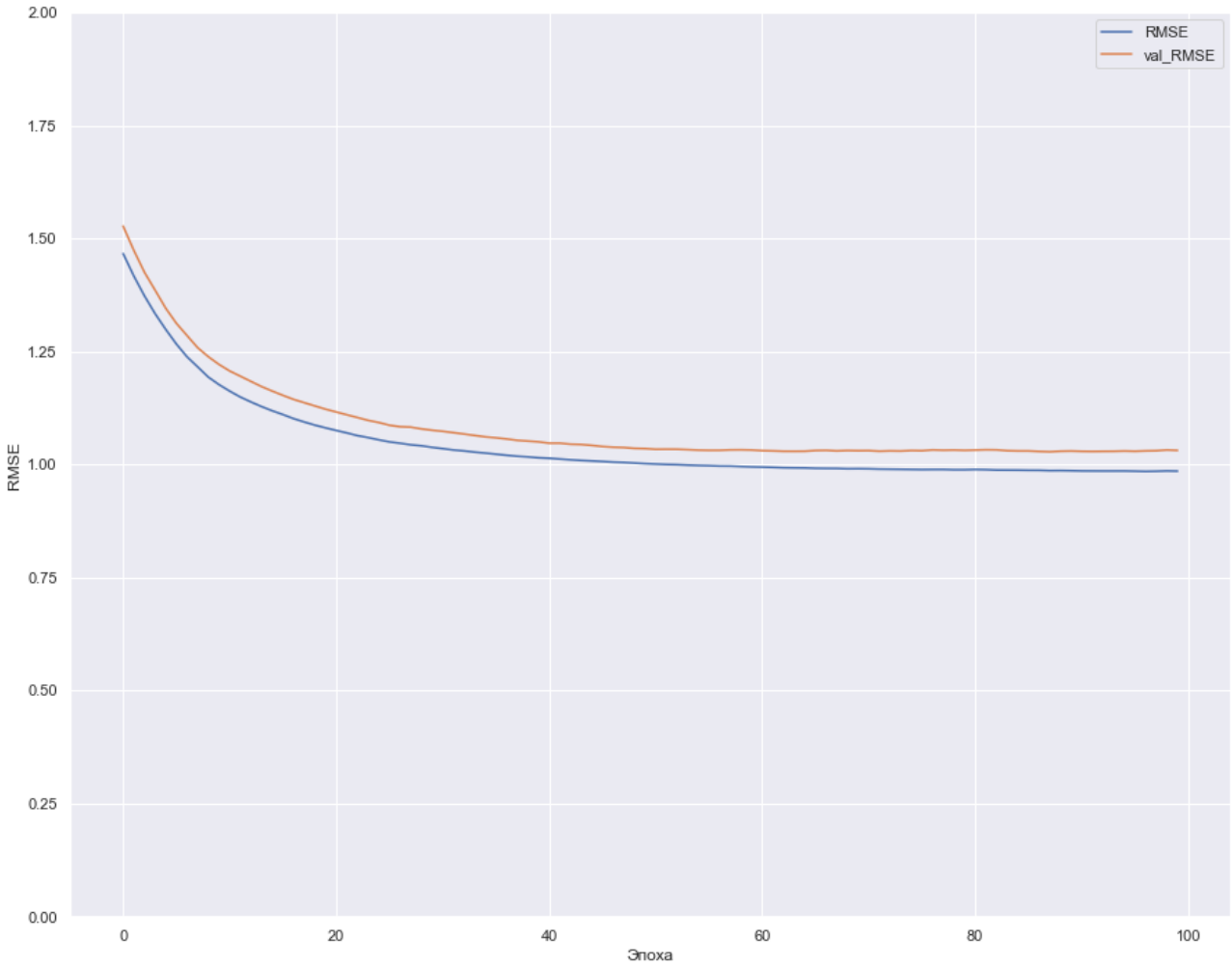

Рисунок 19 — График ошибки обучения нейросети

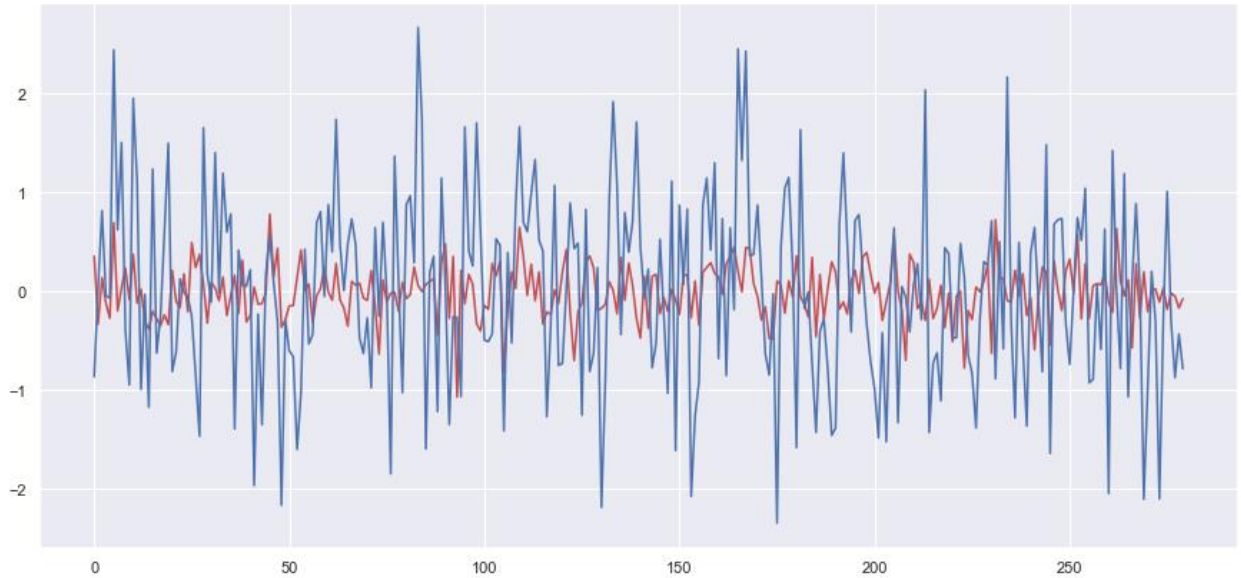

Рисунок 20 – Прогноз сети (красным) и валидационные данные (синим)

Видно, что примерно до 50 эпохи обучение шло хорошо, а потом сеть начала переобучаться. Значение loss на тестовых выборках продолжило уменьшаться, а на валидационной начало расти.

Визуализация результатов показывает, что нейросеть из библиотеки tensorflow старалась подстроиться к данным. Выглядят результаты «похоже», но метрика разочаровывает. По графику делаем вывод что для обучения спроектированной нейронной сети достаточно 50 эпох.

2.5 Тестирование модели

Согласно заданию, необходимо сравнить ошибку каждой модели на тренировочной и тестирующей части выборки. На эту часть у меня не хватило времени, постараюсь до защиты дописать.

2.6. Разработка приложения

Несмотря на то, что пригодных к внедрению моделей получить не удалось, можно разработать тестовое веб-приложение.

В приложении необходимо реализовать следующие функции:

– ввод входных параметров;

– загрузка сохраненной модели;

– получение и отображение прогноза выходных параметров.

Для разработки веб-приложения использовался микрофреймворк Flask. Ввод входных параметров осуществляется через поля ввода на веб странице. Перед подачей в модель входные параметры нормализуются, используя сохранённый в формате pickle нормализатор StandardScaler. Для прогноза загружаем модель нейронной сети Keras. Прогнозное значение масштабируем обратно используя тот же скейлер, что и для входных параметров. Отображаем

данные. Скриншот прогноза соотношения матрица-наполнитель в веб-приложении показан на рисунке 21.

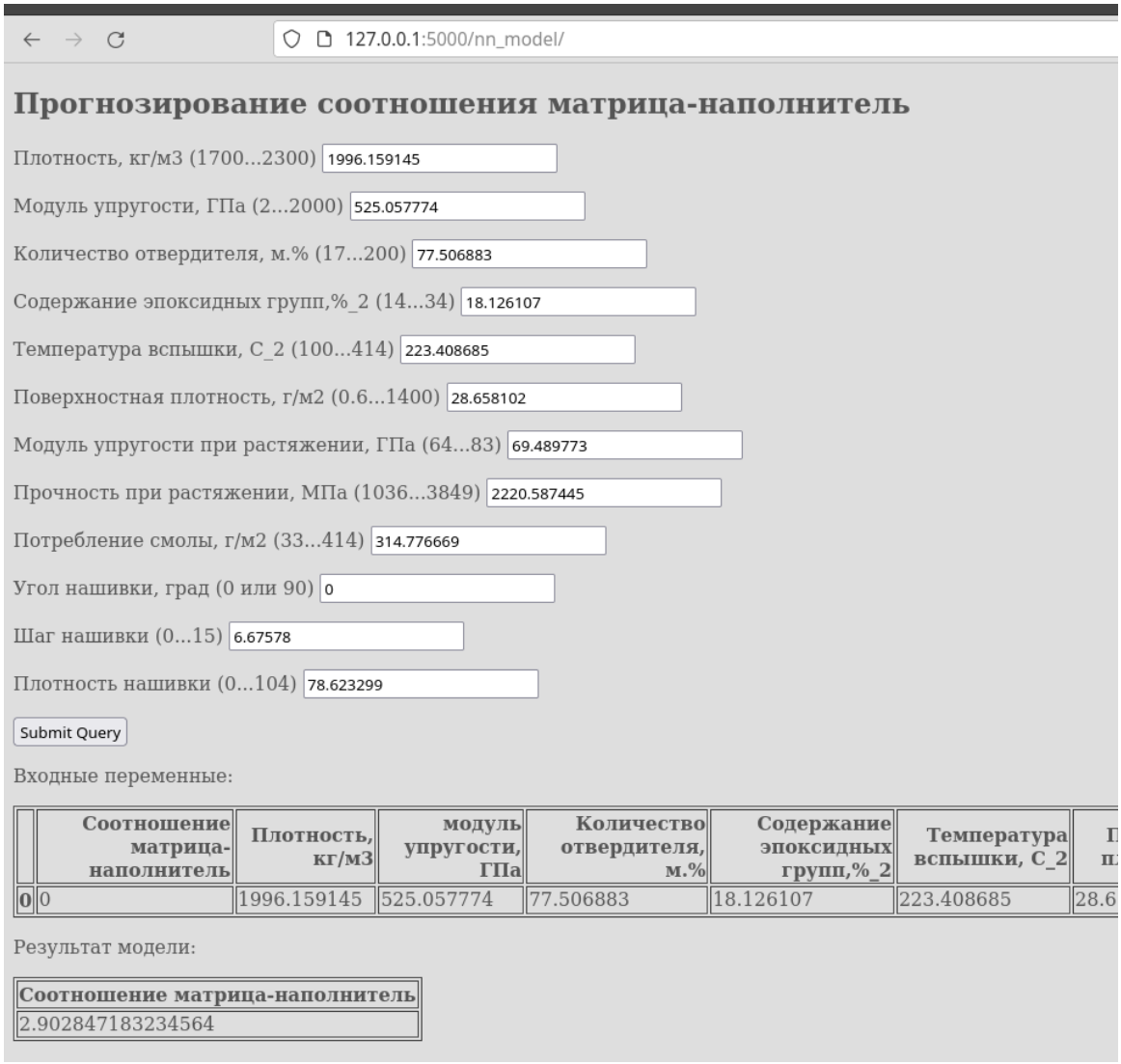

Рисунок 21 – Веб-приложение для прогноза соотношения матрица-наполнитель

2.7. Создание удаленного репозитория

Для данного исследования был создан удаленный репозиторий на GitHub, который находится по адресу https://github.com/E-Mi-Zh/bmstu_ds_thesis . В него загружены результаты работы: исследовательский notebook, код веб-приложения, пояснительная записка и презентация.

Заключение

В ходе выполнения данной работы было выполнено:

– изучение теоретических методов анализа данных и машинного обучения;

– разведочный анализ данных;

– предобработка данных;

– построение регрессионных моделей;

– визуализация модели и оценка качества прогноза;

– сохранение моделей;

– разработка и тестирование веб-приложения.

Возможные причины неудовлетворительной работы моделей:

– нечеткая постановка задачи, отсутствие дополнительной информации о зависимости признаков с точки зрения физики процесса. Незначимые признаки являются для модели шумом, и мешают найти зависимость целевых от значимых входных признаков;

– исследование предварительно обработанных данных. Возможно, на непредобработанных данных можно было бы получить более качественные модели;

– недостаток знаний и опыта, были испробованы не все возможные методы прогноза.

На основании проведенного исследования можно сделать следующие основные выводы по теме:

– распределение полученных данных близко к нормальному;

– коэффициенты корреляции между парами признаков стремятся к нулю;

– примененные модели линейной, полиномиальной регрессии, случайного леса и нейронных сеть не показали высокой эффективности в прогнозировании свойств композитов, необходимы дополнительные вводные данные для улучшения моделей;

– лучшие метрики – критерии качества у моделей случайного леса.

Список использованных источников

1. Силен Дэви, Мейсман Арно, Али Мохамед. Основы Data Science и Big Data. Python и наука о данных. – СПб.: Питер, 2017. – 336 с.: ил.
2. ГрасД. Data Science. Наука о данных с нуля: Пер. с англ. - 2-е изд., перераб. и доп. - СПб.: БХВ-Петербург, 2021. - 416 с.: ил.
3. 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.
4. Библиотека scikit-learn [Электронный ресурс]: – Режим доступа: https://scikit-learn.org (дата обращения: 21.04.2022).
5. Библиотека Keras [Электронный ресурс]: – Режим доступа: https://keras.io/ (дата обращения: 21.04.2022).
6. Библиотека Tensorflow [Электронный ресурс]: – Режим доступа: https://www.tensorflow.org/overview (дата обращения: 21.04.2022).
7. Документация Pandas [Электронный ресурс]: – Режим доступа: https://pandas.pydata.org/docs/user_guide/index.html#user-guide (дата обращения: 21.04.2022).
8. Документация Matplotlib [Электронный ресурс]: – Режим доступа: https://matplotlib.org/ (дата обращения: 21.04.2022).
9. Документация Seaborn [Электронный ресурс]: – Режим доступа: https://seaborn.pydata.org/ (дата обращения: 21.04.2022).
10. Документация Flask [Электронный ресурс]: – Режим доступа: https://pypi.org/project/Flask/ (дата обращения: 21.04.2022).