ЦЕНТР ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ МГТУ им. Н.Э. Баумана

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

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 Карпов Филипп Алексеевич

Москва, 2024

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

СОДЕРЖАНИЕ

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

Введение

Тема работы – прогнозирование конечных свойств новых материалов (композиционных материалов).

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

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

НТИ «Цифровое материаловедение: новые материалы и вещества» (структурное подразделение МГТУ им. Н.Э. Баумана).

Созданные прогнозные модели помогут сократить количество проводимых испытаний, а также пополнить базу данных материалов возможными новыми характеристиками материалов и цифровыми двойниками новых композитов.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

1. Аналитическая часть

1.1 Постановка задачи

Для исследовательской работы были даны 2 файла: X_bp.xlsx (с данными о параметрах базальтопластика, состоящий из 1023 строк и 10 столбцов данных) и X_nup.xlsx (данными углепластика, 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.

Практическая часть работы будет реализована на языке Python. Далее по тексту упомянутые используемые метод будут относиться к библиотекам в Python.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

1.2 Описание методов машинного обучения

Данная задача в рамках классификации методов машинного обучения относится к машинному обучению с учителем, так как в предоставленном наборе данных имеются значения целевых параметров.

Так как перед нами стоит задача предсказания значений вещественной переменной — это задача регрессии.

В настоящее время разработано много методов регрессионного анализа. В данной работе были исследованы (и некоторые из них применены) следующие методы:

1) линейная регрессия (Linear regression);
2) Метод опорных веторов
3) случайный лес (Random Forest);
4) К-ближайших соседей (KNeighbors Regressor);
5) градиентный бустинг (Gradient Boosting Regressor).

1.2.1 Линейная регрессия

Простая линейная регрессия имеет место, если рассматривается зависимость между одной входной и одной выходной переменными. Для этого определяется уравнение регрессии (1) и строится соответствующая прямая, известная как линия регрессии.

$$y = ax + b \quad (1)$$

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.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

Если ищется зависимость между несколькими входными и одной выходной переменными, то имеет место множественная линейная регрессия. Соответствующее уравнение имеет вид (2).

$$Y = b0 + b1 * x1 + b2 * x2 + \cdots + bn * xn, \quad (2)$$

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.

Линейная регрессия — первый тщательно изученный метод регрессионного анализа. Его главное достоинство — простота. Такую модель можно построить и рассчитать даже без мощных вычислительных средств. Простота является и главным недостатком этого метода. Тем не менее, именно с линейной регрессии целесообразно начать подбор подходящей модели.

1.2.2 Метод опорных векторов

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.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

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.

1.2.3 Случайный лес

Случайный лес (RandomForest) — представитель ансамблевых методов.

Если точность дерева решений оказалось недостаточной, мы можем множество моделей собрать в коллектив. Формула итогового решателя (9) — это усреднение предсказаний отдельных деревьев.

$$a(x) = \frac{1}{N}\sum_{i=1}^{N} b_i(x) \tag{9}$$

где

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

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 Метод К-случайных соседей

Метод К-ближайших соседей (k Nearest Neighbors) – это метод классификации, который адаптирован для регрессии. На интуитивном уровне суть метода проста: посмотри на соседей вокруг, какие из них преобладают, таковым ты и являешься.

В случае использования метода для регрессии, объекту присваивается среднее значение по k ближайшим к нему объектам, значения которых уже известны.

Для реализации метода необходима метрика расстояния между объектами. Используется, например, эвклидово расстояние для количественных признаков или расстояние Хэмминга для категориальных.

Этот метод — пример непараметрической регрессии.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

1.2.5 Градиентный бустинг

Градиентный бустинг (GradientBoosting) — еще один представитель ансамблевых методов.

В отличие от случайного леса, где каждый базовый алгоритм строится независимо от остальных, бустинг воплощает идею последовательного построения линейной комбинации алгоритмов. Каждый следующий алгоритм старается уменьшить ошибку предыдущего.

Чтобы построить алгоритм градиентного бустинга, нам необходимо выбрать базовый алгоритм и функцию ошибки (loss). Loss-функция – это мера, которая показывает насколько хорошо предсказание модели соответствуют данным. Используя градиентный спуск и обновляя предсказания, основанные на скорости обучения (learning rate), ищем значения, на которых функция ошибки минимальна.

Бустинг, использующий деревья решений в качестве базовых алгоритмов, называется градиентным бустингом над решающими деревьями. Он отлично работает на выборках с «табличными», неоднородными данными и способен эффективно находить нелинейные зависимости в данных различной природы. На настоящий момент это один из самых эффективных алгоритмов машинного обучения. Благодаря этому он широко применяется во многих конкурсах и промышленных задачах. Он проигрывает только нейросетям на однородных данных (изображения, звук и т. д.).

Из недостатков алгоритма можно отметить только затраты времени на вычисления и необходимость грамотного подбора гиперпараметров.

1.2.6 Нейронная сеть

Нейронная сеть — это последовательность нейронов, соединенных между собой связями. Структура нейронной сети пришла в мир

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

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, гиперболический тангенс.

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 определяет 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. Веса каждого соединения обновляются таким образом, чтобы минимизировать значение функции потерь.

Для обновления весов в модели используются различные оптимизаторы.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

Количество эпох показывает, сколько раз выполнялся проход для всех примеров обучения.

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.

2 Разведочный анализ данных

Прежде всего необходимо изучить датасет и выявить его основные характеристики.

В итоговом (объединенном из двух файлов) датасете имеется 1023 объекта с 13ю признаками, 3 из которых будут выступать в качестве целевой переменной (входных данных). На рисунке 1 можно видеть заголовки и первые 5 строк датасета.

Соотношение матрица-наполнитель	Плотность, кг/м3	модуль упругости, ГПа	Количество отвердителя, м.%	Содержание эпоксидных групп,%_2	Температура вспышки, С_2	Поверхностная плотность, г/м2	Модуль упругости при растяжении, ГПа	Прочность при растяжении, МПа	Потребление смолы, г/м2	Угол нашивки, град	Шаг нашивки	Плотность нашивки
1.857143	2030.0	738.736842	30.00	22.267857	100.000000	210.0	70.0	3000.0	220.0	0	4.0	57.0
1.857143	2030.0	738.736842	50.00	23.750000	284.615385	210.0	70.0	3000.0	220.0	0	4.0	60.0
1.857143	2030.0	738.736842	49.90	33.000000	284.615385	210.0	70.0	3000.0	220.0	0	4.0	70.0
1.857143	2030.0	738.736842	129.00	21.250000	300.000000	210.0	70.0	3000.0	220.0	0	5.0	47.0
2.771331	2030.0	753.000000	111.86	22.267857	284.615385	210.0	70.0	3000.0	220.0	0	5.0	57.0

Рисунок 1 Заголовки и первые 5 строк датасета

В первой части работы мы будем прогнозировать Модуль упругости при растяжении и Прочность при растяжении, за входные данные будем брать остальные 11 признаков. Для прогнозирования каждой целевой переменной будет подбираться своя модель.

Во второй части работы займемся прогнозированием Соотношения матрица-наполнитель с помощью нейронных сетей. За входные признаки будут взяты остальные 12 характеристик композитных материалов из датасета.

Было установлено, что все характеристики являются числовыми (12 признаков с вещественными числами и один с целыми), пропусков в данных нет (см. рисунок 2).

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

```
#Смотрим информацию о датасете
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1023 entries, 0 to 1022
Data columns (total 13 columns):
 #   Column                            Non-Null Count  Dtype  
---  ------                            --------------  -----  
 0   Соотношение матрица-наполнитель   1023 non-null   float64
 1   Плотность, кг/м3                  1023 non-null   float64
 2   модуль упругости, ГПа             1023 non-null   float64
 3   Количество отвердителя, м.%       1023 non-null   float64
 4   Содержание эпоксидных групп,%_2   1023 non-null   float64
 5   Температура вспышки, С_2          1023 non-null   float64
 6   Поверхностная плотность, г/м2     1023 non-null   float64
 7   Модуль упругости при растяжении, ГПа  1023 non-null   float64
 8   Прочность при растяжении, МПа     1023 non-null   float64
 9   Потребление смолы, г/м2           1023 non-null   float64
 10  Угол нашивки, град                1023 non-null   int64  
 11  Шаг нашивки                       1023 non-null   float64
 12  Плотность нашивки                 1023 non-null   float64
dtypes: float64(12), int64(1)
memory usage: 111.9 KB
```

Рисунок 2 Информация о датасете

Кроме того, было установлено, что в основном объекты имеют различные значения признаков, за исключением Угла нашивки. Это хорошо видно на рисунке 3. Однако, учитывая физический смысл величины, попробуем оставить этот признак в неизмененном виде.

```
#Смотрим количество уникальных значений в каждом столбце
df.nunique()
```

```
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
dtype: int64
```

Рисунок 3 Количество уникальных значений в каждом столбце

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

Цель разведочного анализа данных — выявить закономерности в данных. Для корректной работы большинства моделей желательна сильная зависимость целевых переменных от входных и отсутствие зависимости между входными переменными.

В качестве инструментов разведочного анализа используется: оценка статистических характеристик датасета (рисунок 4); гистограммы распределения каждой из переменной, диаграммы ящика с усами (рисунок 5); попарные графики рассеяния точек (рисунок 5); тепловая карта (несколько вариантов); описательная статистика для каждой переменной; анализ и полное исключение выбросов; проверка наличия пропусков и дубликатов; ранговая корреляция Кендалла и Пирсона.

	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 4 Описательная статистика датасета

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

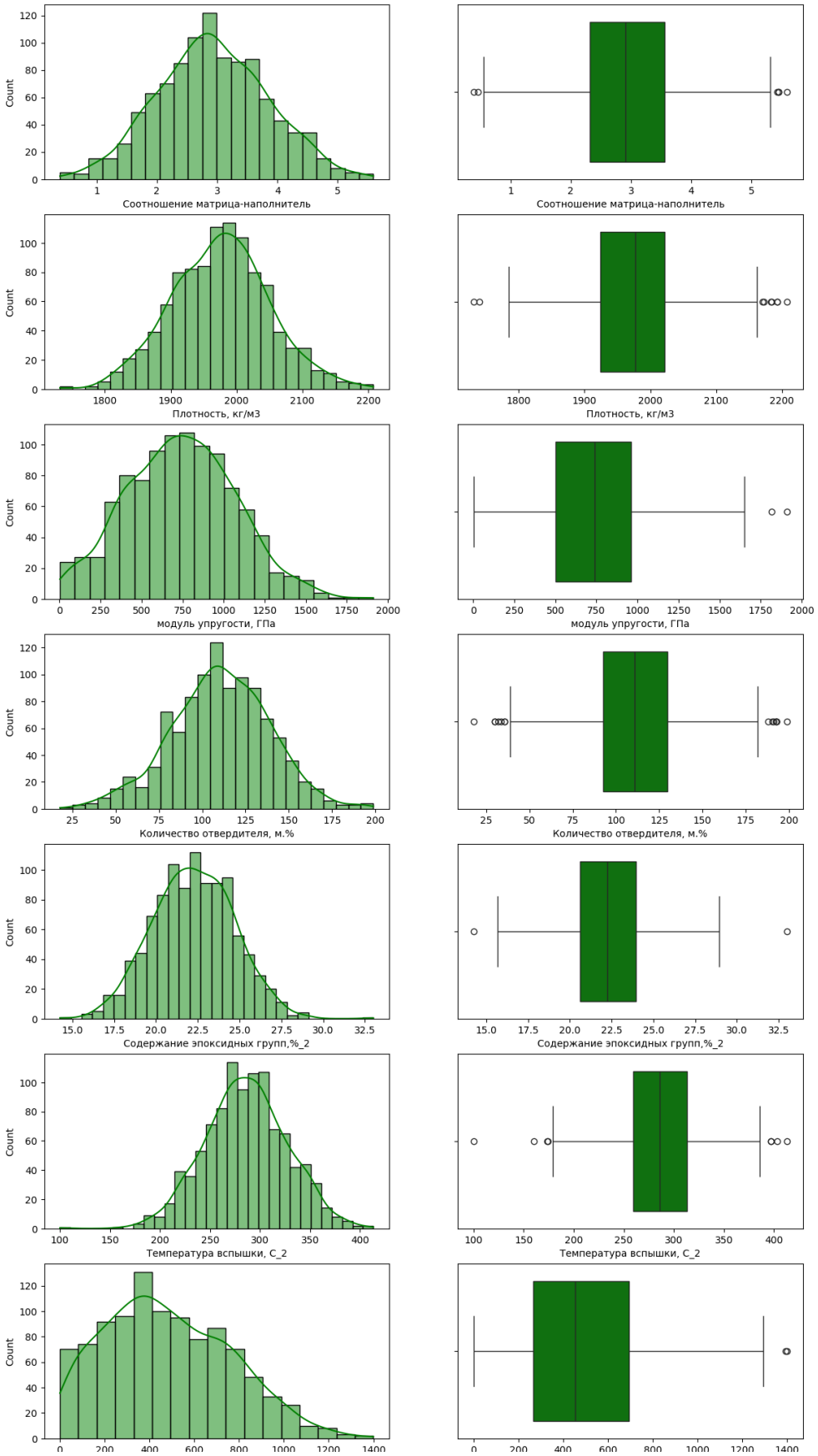

Рисунок 5 Графики распределения величин и диаграммы «ящик с усами»

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

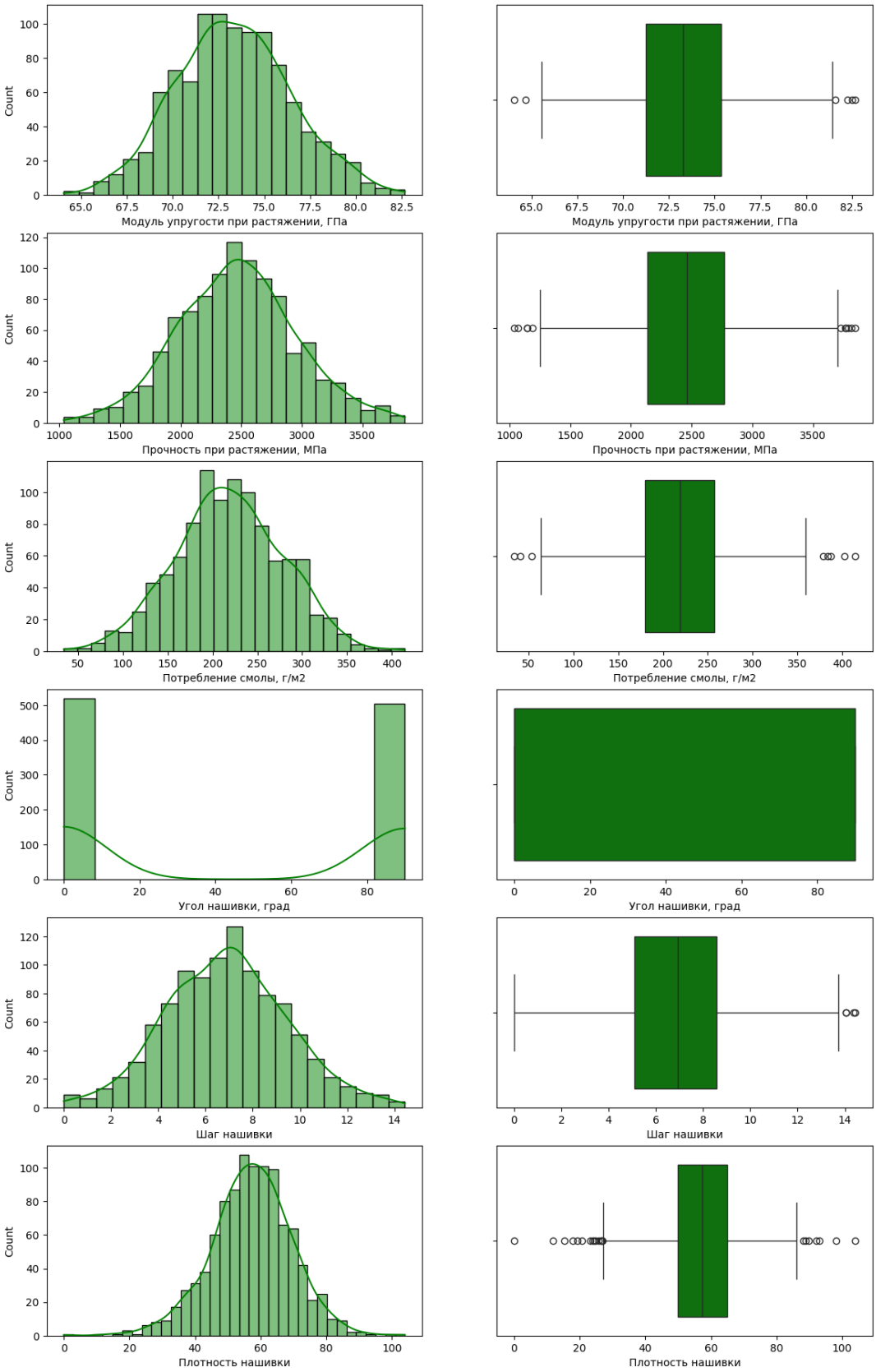

Рисунок 5 Графики распределения величин и диаграммы «ящик с усами» (продолжение)

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

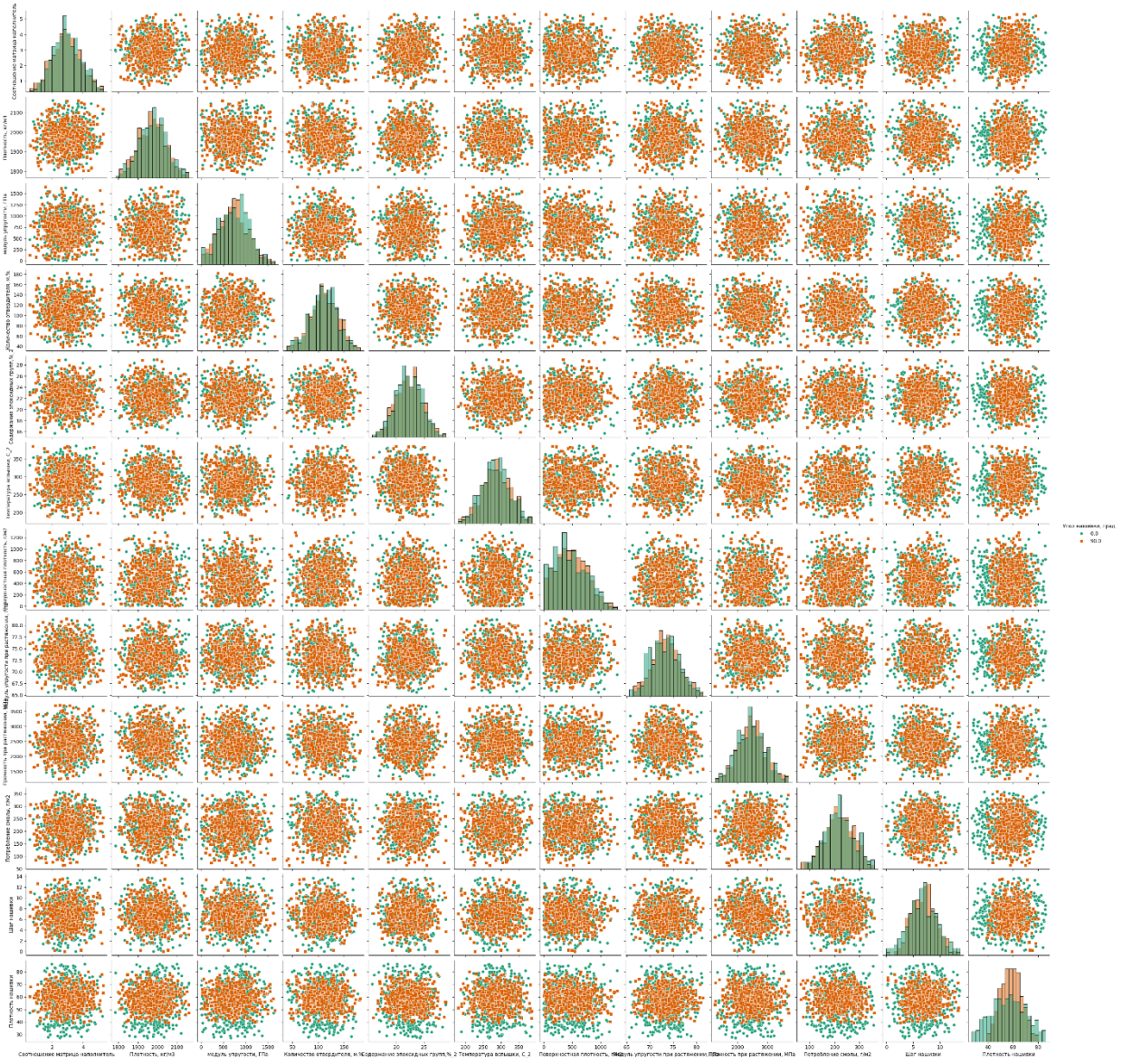

Рисунок 6 График попарного рассеяния точек

На рисунке 6 мы видим график попарного рассеяния точек. По форме «облаков точек» не видно каких-либо закономерностей, что означает отсутствие линейных и других зависимостей, похожих на какую-либо известную функцию, между парами признаков (простыми словами, «облака точек» не стремятся к прямой, гиперболе, экспоненте и т.п.). Кроме того, очевидно наличие выбросов (об этом говорят достаточно удаленные точки от общего «облака точек»).

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

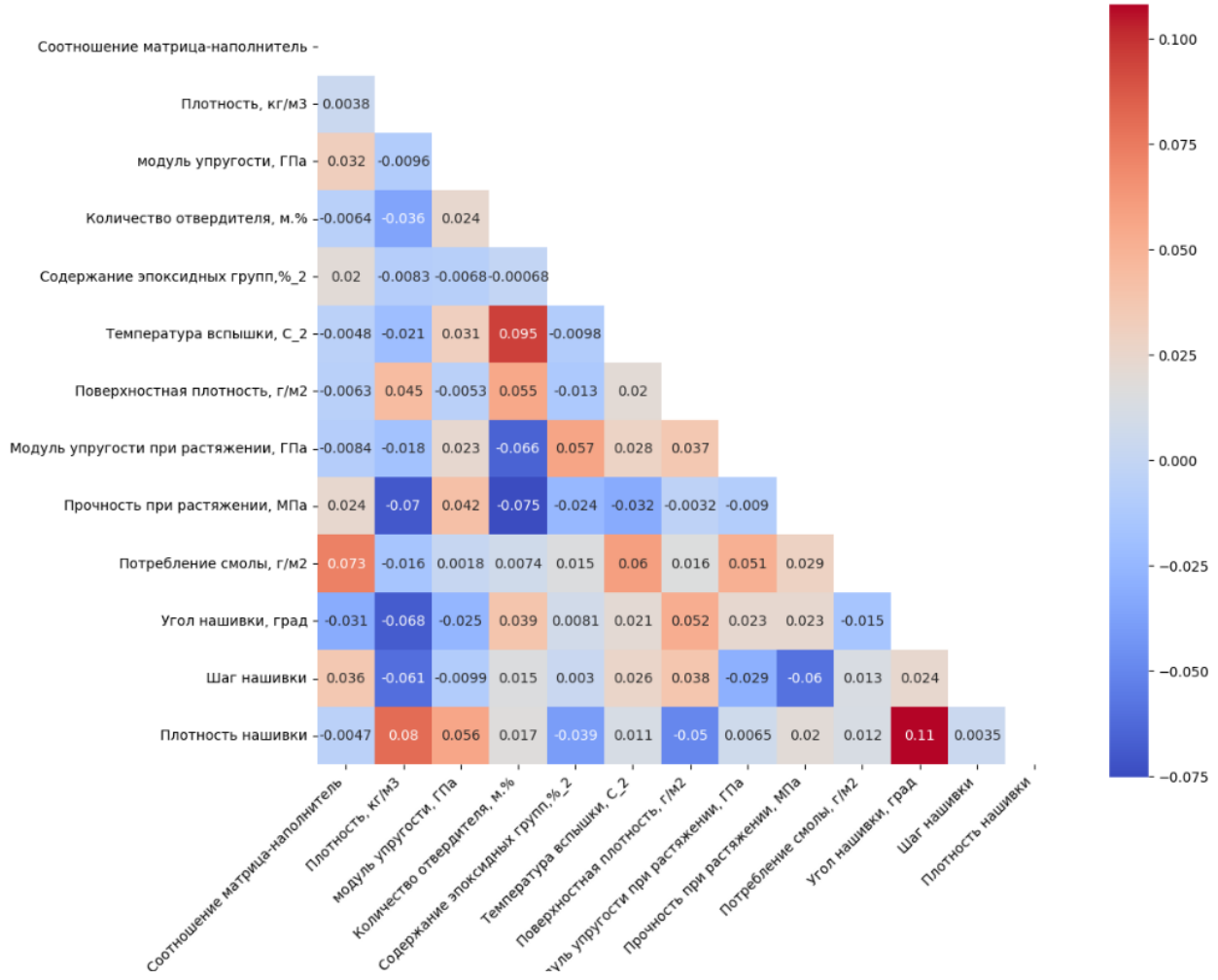

Рисунок 7 Тепловая карта корреляции

Коэффициент корреляции в математической статистике – это показатель, характеризующий силу статистической связи между двумя или несколькими случайными величинами.

Значения коэффициента корреляции всегда расположены в диапазоне от -1 до 1 и интерпретируются следующим образом:

- если коэффициент корреляции близок к 1, то между переменными наблюдается положительная корреляция. Иными словами, отмечается высокая степень связи между переменными. Если значения одной переменной будут возрастать, то вторая переменная будет увеличиваться;

- если коэффициент корреляции близок к -1, это означает, что между переменными имеет место сильная отрицательная корреляция. Иными

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

словами, так же отмечается высокая степень связи между переменными. Но если значения одной переменной будут возрастать, то вторая переменная будет уменьшаться;

- промежуточные значения, близкие к 0, указывают на слабую корреляцию между переменными и, соответственно, низкую зависимость. Иными словами, поведение одной переменной не будет совсем (или почти совсем) влиять на поведение другой.

Очевидно, что если корреляция между переменными высокая, то, зная поведение входной переменной, проще предсказать поведение выходной, и полученное предсказание будет точнее (говорят, что входная переменная хорошо «объясняет» выходную).

Простой коэффициент корреляции (а здесь мы говорили именно о нем) Пирсона описывает только степень линейной связи и применим к непрерывным величинам.

По нашей матрице корреляции мы видим, что все коэффициенты корреляции близки к нулю, что означает отсутствие линейной зависимости между признаками. Это сразу говорит о том, что найти хорошо работающую модель для решения данной задачи будет сложно.

После обнаружения выбросов данные, значительно отличающиеся от выборки, будут полностью удалены. Для расчёта этих данных мы будем использовать метод межквартильного расстояния. После удаления выбросов графики приобрели следующий вид (

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

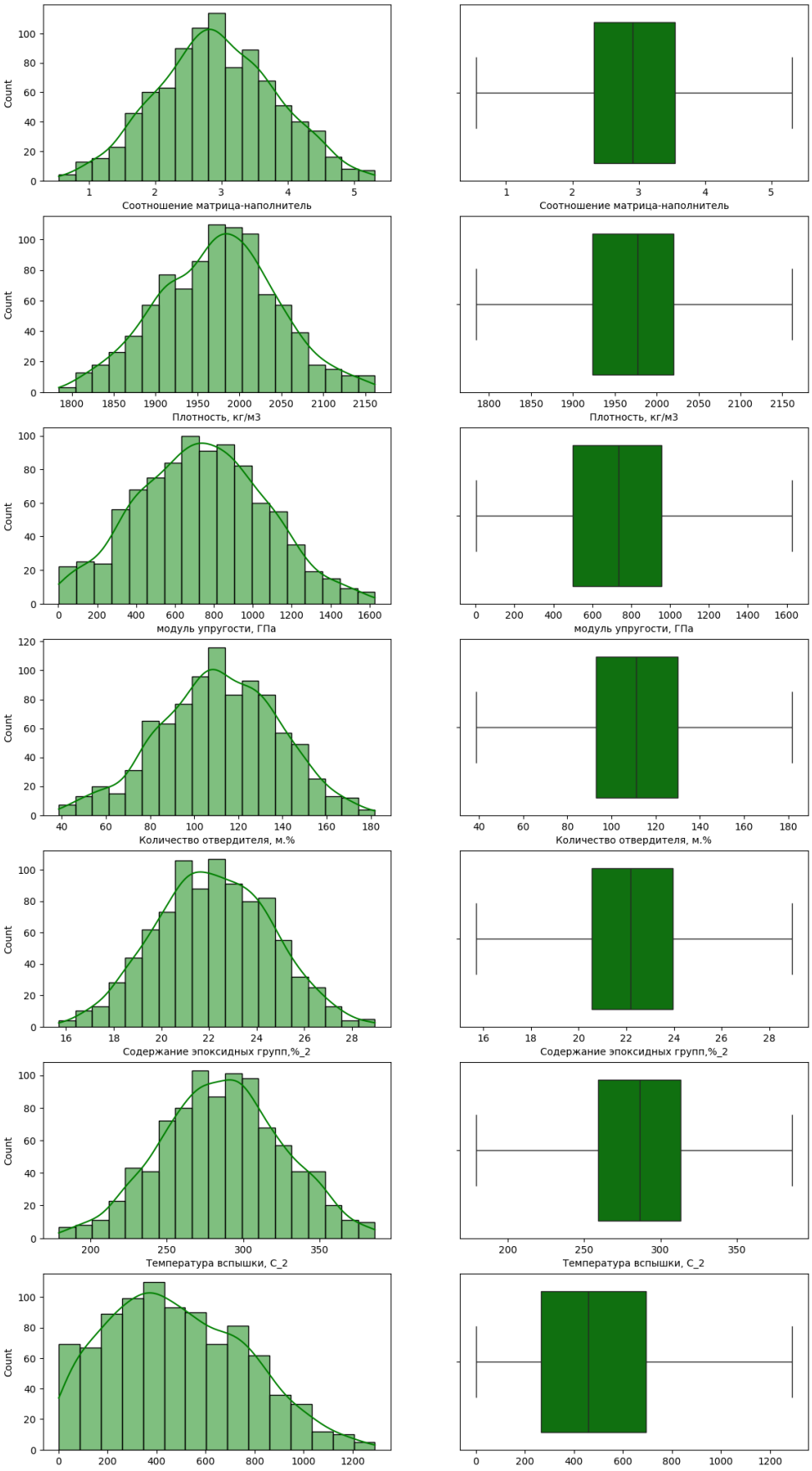

Рисунок 7 Графики распределения величин и диаграммы «ящик с усами» после удаления выбросов

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

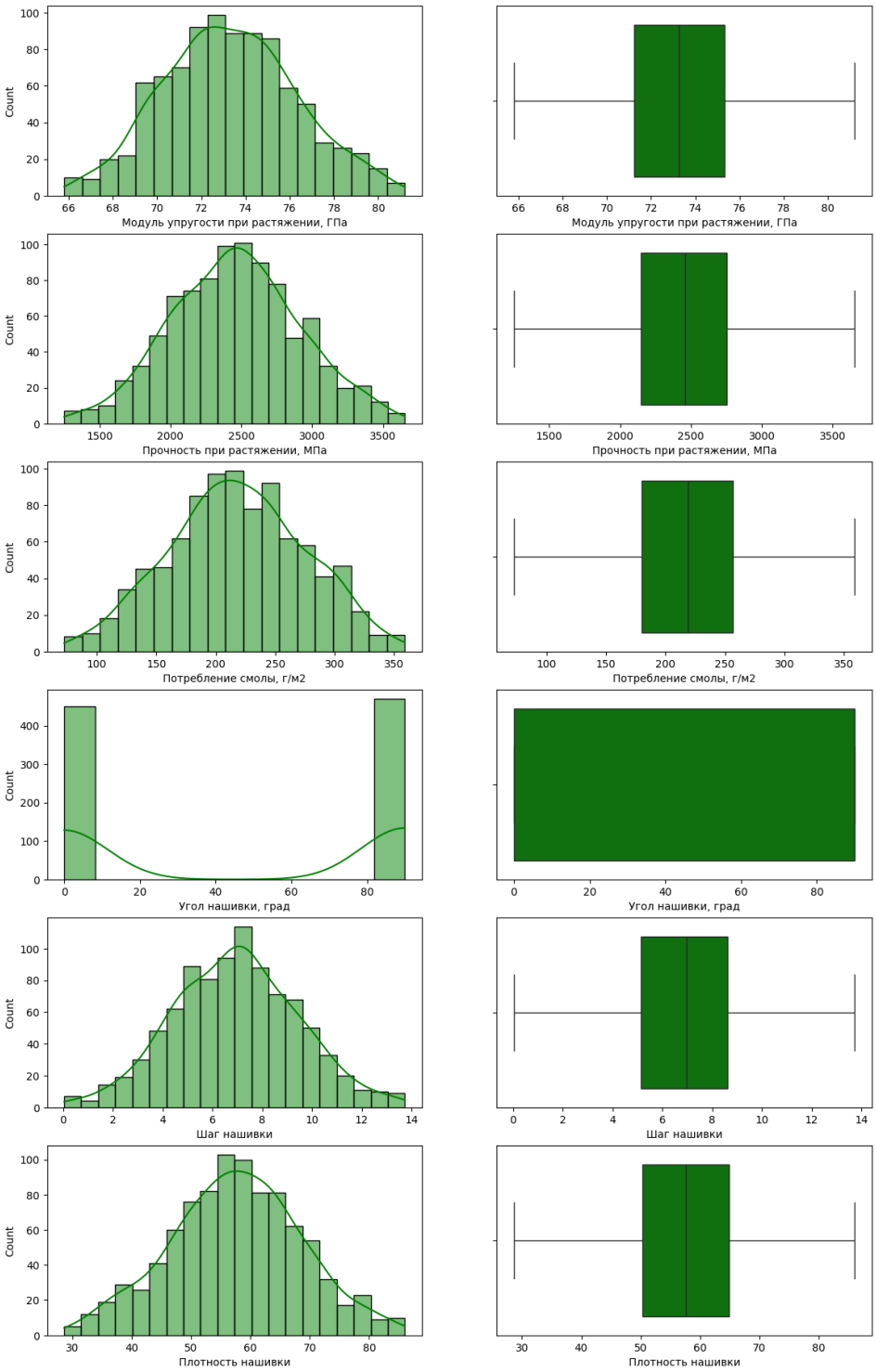

Рисунок 7 Графики распределения величин и диаграммы «ящик с усами» после удаления выбросов (продолжение)

В данной работе датасет был полностью нормализован с помощью метода MinMaxScaler библиотеки sklearn. Этот метод преобразует каждый

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

признак по отдельности, так, чтобы его значения находились в диапазоне от 0 до 1. На рисунке 9 можно увидеть описательную статистику данных после нормализации. Видно, что минимальные значения по всем признакам теперь 0, максимальные – 1.

	count	mean	std	min	25%	50%	75%	max
Соотношение матрица-наполнитель	922.0	0.499412	0.187858	0.0	0.371909	0.495189	0.629774	1.0
Плотность, кг/м3	922.0	0.502904	0.188395	0.0	0.368184	0.511396	0.624719	1.0
модуль упругости, ГПа	922.0	0.451341	0.201534	0.0	0.305188	0.451377	0.587193	1.0
Количество отвердителя, м.%	922.0	0.506200	0.186876	0.0	0.378514	0.506382	0.638735	1.0
Содержание эпоксидных групп,%_2	922.0	0.490578	0.180548	0.0	0.366571	0.488852	0.623046	1.0
Температура вспышки, С_2	922.0	0.516739	0.190721	0.0	0.386228	0.516931	0.646553	1.0
Поверхностная плотность, г/м2	922.0	0.373295	0.217269	0.0	0.204335	0.354161	0.538397	1.0
Модуль упругости при растяжении, ГПа	922.0	0.487343	0.196366	0.0	0.353512	0.483718	0.617568	1.0
Прочность при растяжении, МПа	922.0	0.503776	0.188668	0.0	0.373447	0.501481	0.624299	1.0
Потребление смолы, г/м2	922.0	0.507876	0.199418	0.0	0.374647	0.510143	0.642511	1.0
Угол нашивки, град	922.0	0.510846	0.500154	0.0	0.000000	1.000000	1.000000	1.0
Шаг нашивки	922.0	0.503426	0.183587	0.0	0.372844	0.506414	0.626112	1.0
Плотность нашивки	922.0	0.503938	0.193933	0.0	0.376869	0.504310	0.630842	1.0

Рисунок 8 Описательная статистика после нормализации

Также провели стандартизацию данных (рисунок 9)

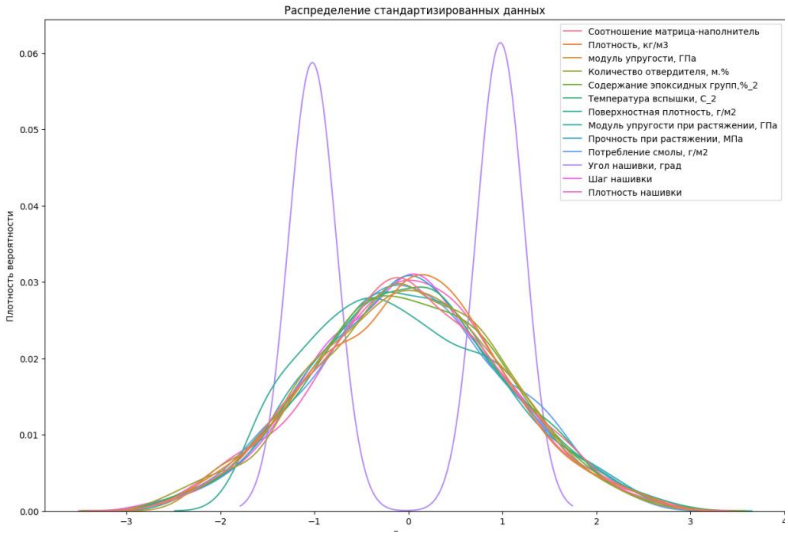

Рисунок 9 Распределение данных после проведения стандартизации

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

3 Разработка, обучение и тестирование моделей

В данной работе рассматривались несколько моделей для каждой задачи и для сравнения моделей между собой, а также для оценки качества работы каждой модели необходимо определить метрики.

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

– 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;

– MAE (Mean Absolute Error) - средняя абсолютная ошибка, принимает значениях в тех же единицах, что и целевая переменная, по ней можно понять абсолютное значение, на которое модель ошибается;

– MSE (Mean Squared Error) - средняя квадратичная ошибка, её нельзя никак интерпретировать. Её можно только сравнить со среднеквадратичной ошибкой другой модели. Т.е. это только способ сравнить 2 модели между собой.

– RMSE (Root Mean Square Error, среднеквадратичная ошибка) — это метрика, используемая для оценки качества моделей, предсказывающих числовые значения. Она измеряет среднюю величину ошибки между предсказанными и фактическими значениями, причём большие ошибки "наказываются" сильнее, так как они возводятся в квадрат.

Целевой признак – Модуль упругости при растяжении/Прочность при растяжении.

На вход моделям подавались 10 признаков:

- Соотношение матрица-наполнитель;

- Плотность, кг/м3;

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

- Количество отвердителя, м.%;
- Содержание эпоксидных групп, %_2;
- Температура вспышки, С_2;
- Поверхностная плотность, г/м2;
- Потребление смолы, г/м2;
- Угол нашивки, град;
- Шаг нашивки;
- Плотность нашивки.

Исследовались следующие модели:

- Метод опорных векторов
- Линейная регрессия (метод LinearRegression библиотеки sklearn);
- Метод К-ближайших соседей (метод KNeighborsRegressor библиотеки sklearn);
- Случайный лес;
- Градиентный бустинг (метод GradientBoostingRegressor библиотеки sklearn)

Для подбора гипермараметров использовался инструмент GridSearchCV.

GridSearchCV позволяет перебирать значения гиперпараметров на заданной сетке и искать лучшие параметры с помощью кросс-валидации.

	Модель	MSE	RMSE	MAE	R2
0	Метод опорных векторов	0.969	0.985	0.792	-0.009
1	Линейная регрессия	0.038	0.194	0.156	-0.016
2	Метод К-ближайших соседей	0.037	0.192	0.154	0.002
3	Случайный лес	0.038	0.195	0.155	-0.025
4	Градиентный бустинг	0.037	0.193	0.154	-0.004

Рисунок 10 Сравнение моделей для прогнозирования Модуля упругости при растяжении

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

	Модель	MSE	RMSE	MAE	R2
0	Метод опорных векторов	0.981	0.99	0.799	-0.001
1	Линейная регрессия	0.035	0.186	0.15	0.009
2	Метод К-ближайших соседей	0.035	0.188	0.152	-0.01
3	Случайный лес	0.035	0.186	0.149	0.008
4	Градиентный бустинг	0.035	0.188	0.152	-0.013

Рисунок 11 Сравнение моделей для прогнозирования прочности при растяжении

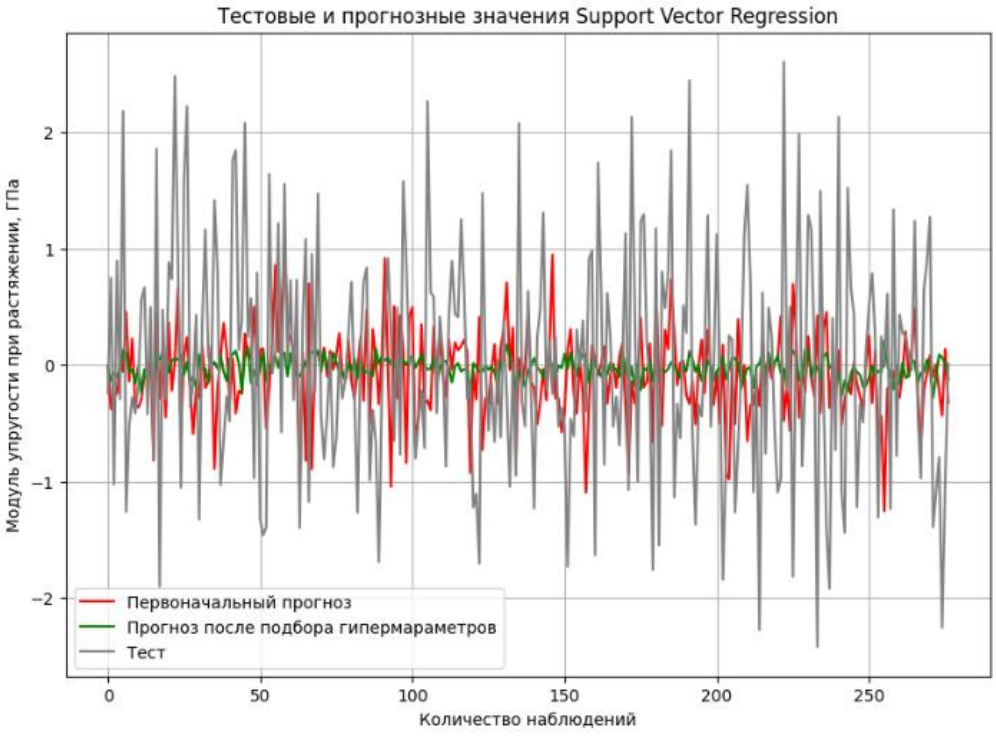

Рисунок 12 Результаты прогнозирования модуля упругости при растяжении методом опорных векторов

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

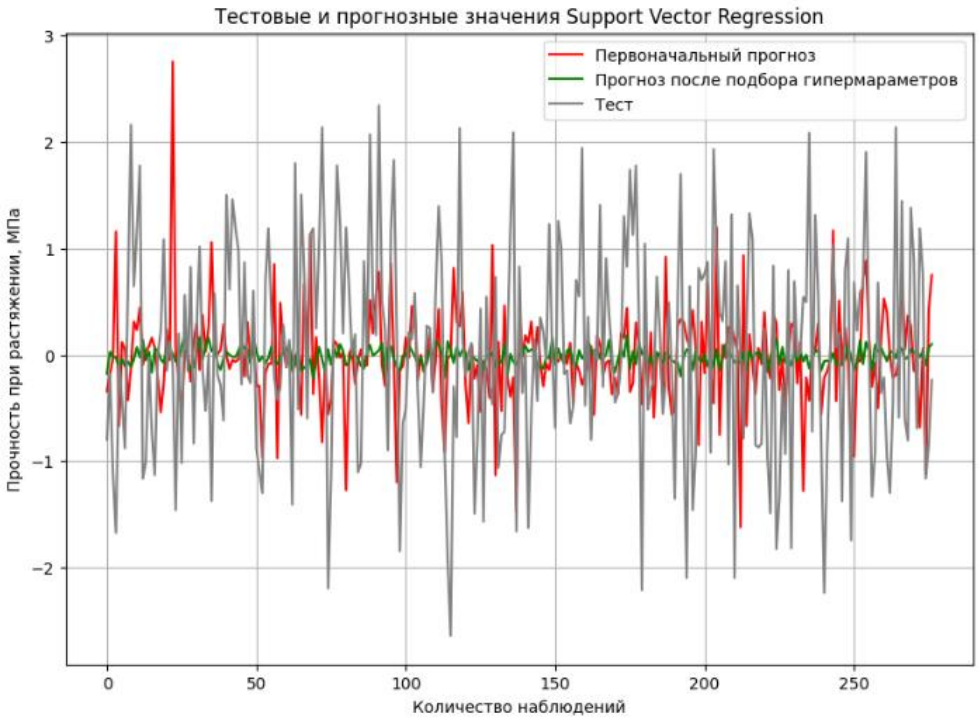

Рисунок 13 Результаты прогнозирования прочности при растяжении методом опорных векторов

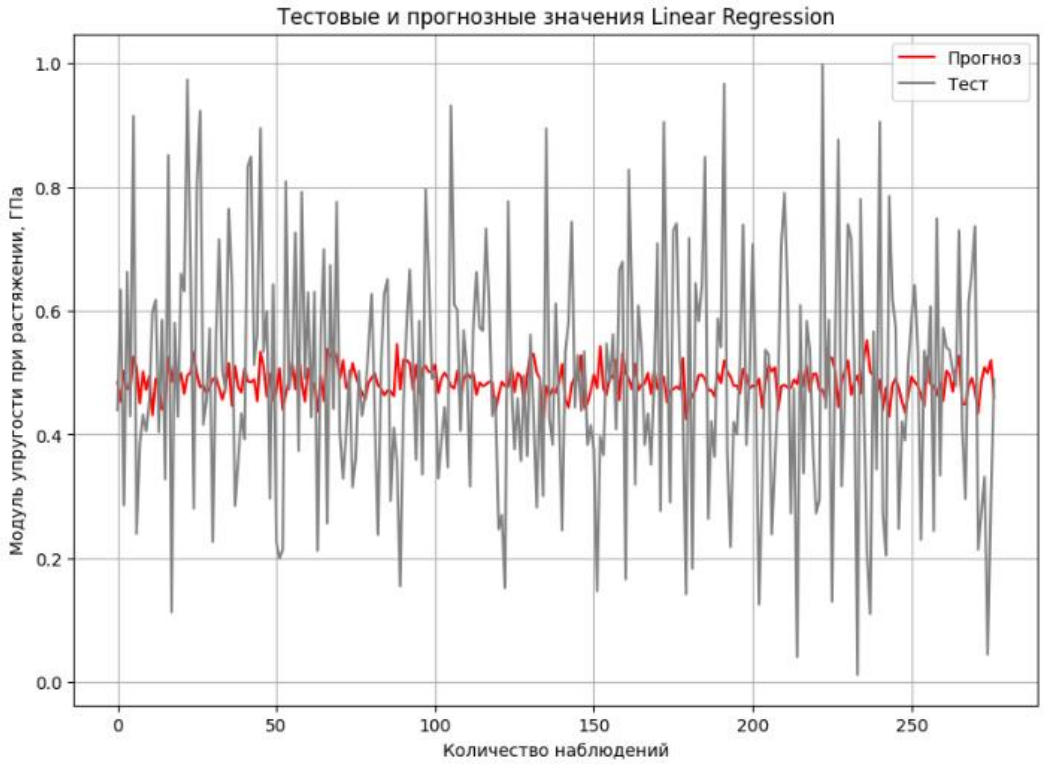

Рисунок 14 Результаты прогнозирования модуля упругости при растяжении методом линейной регрессии

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

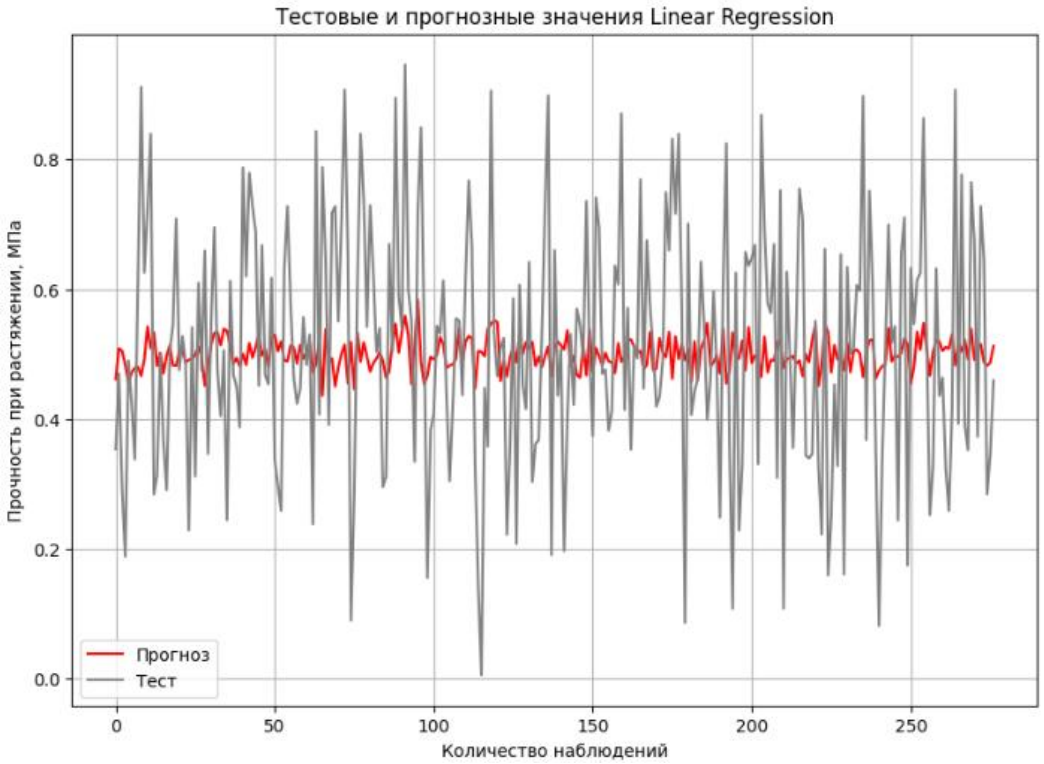

Рисунок 15 Результаты прогнозирования прочности при растяжении методом линейной регрессии

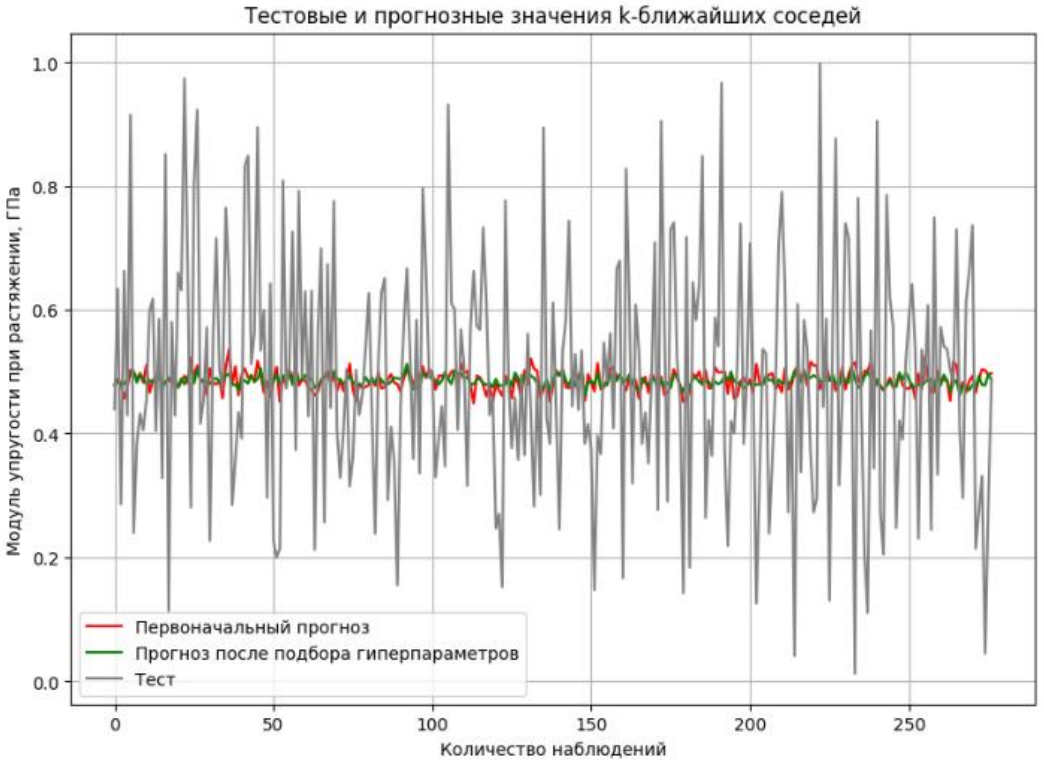

Рисунок 16 Результаты прогнозирования модуля упругости при растяжении методом k-ближайших соседей

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

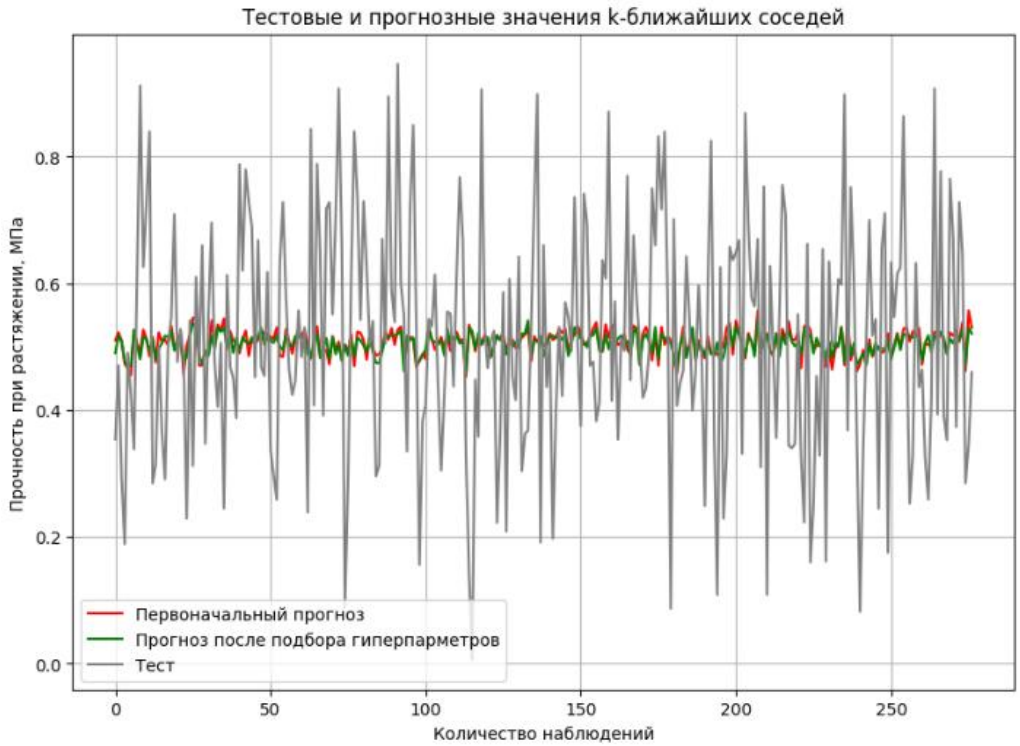

Рисунок 17 Результаты прогнозирования прочности при растяжении методом k-ближайших соседей

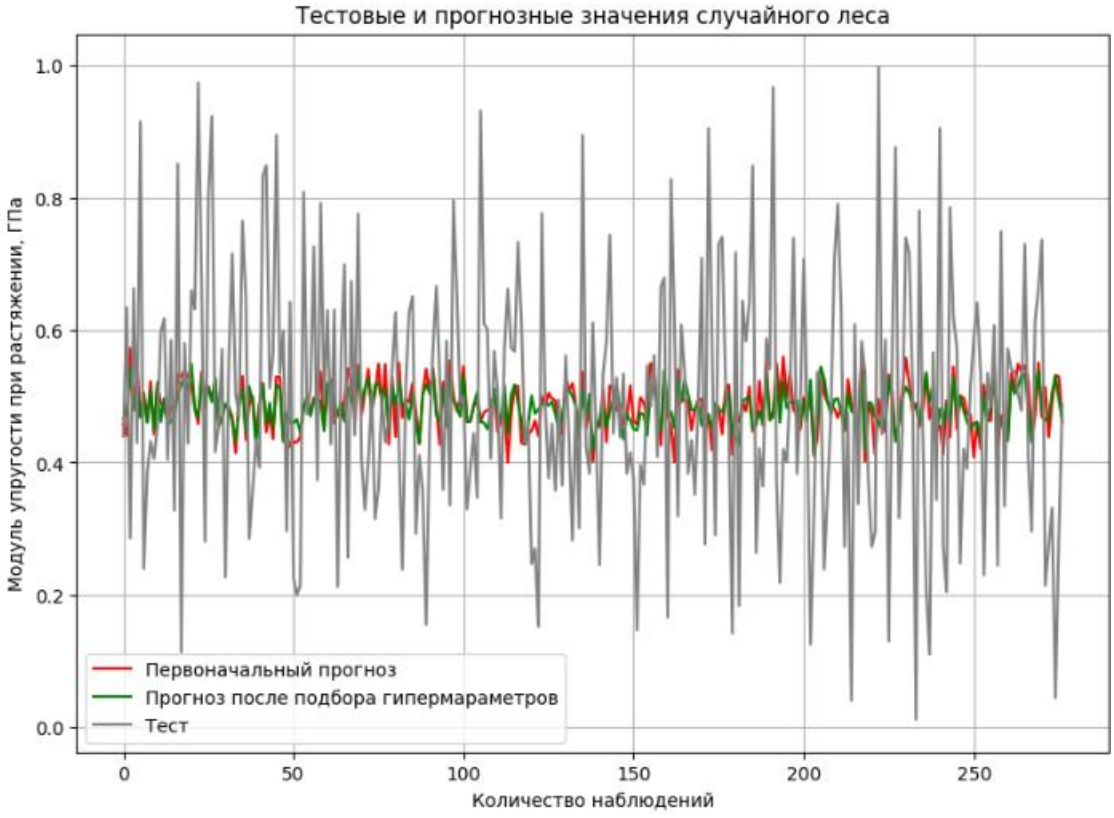

Рисунок 18 Результаты прогнозирования модуля упругости при растяжении методом случайного леса

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

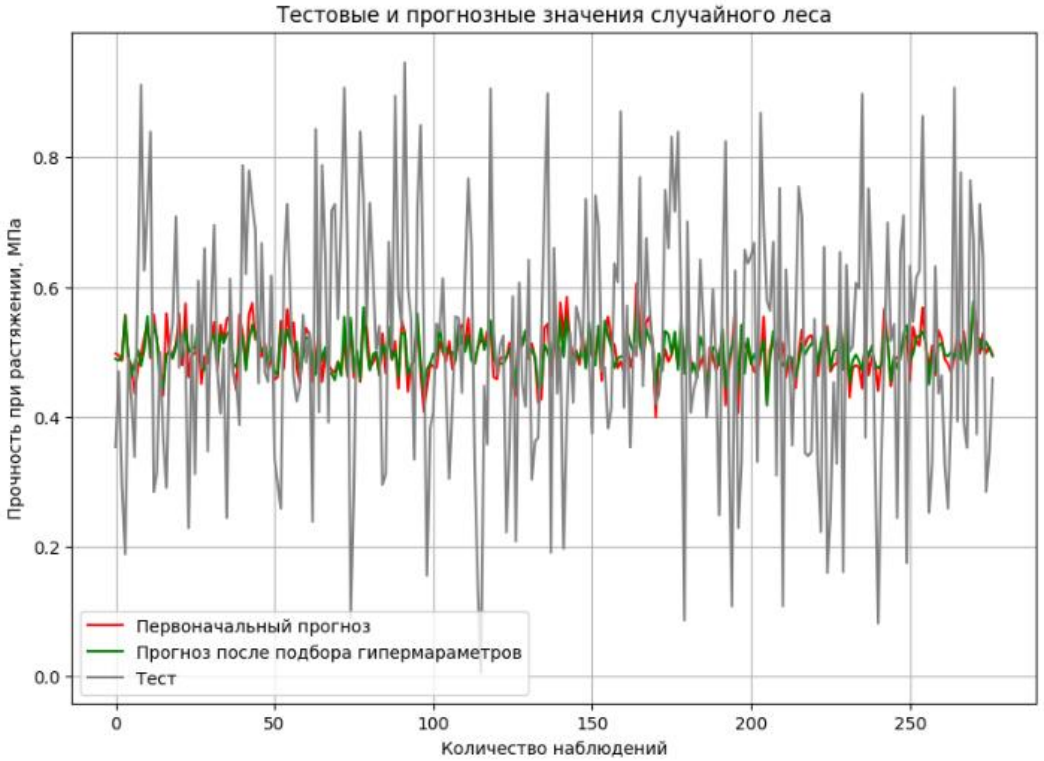

Рисунок 19 Результаты прогнозирования прочности при растяжении методом случайного леса

Рисунок 20 Результаты прогнозирования модуля упругости при растяжении методом градиентного бустинга

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

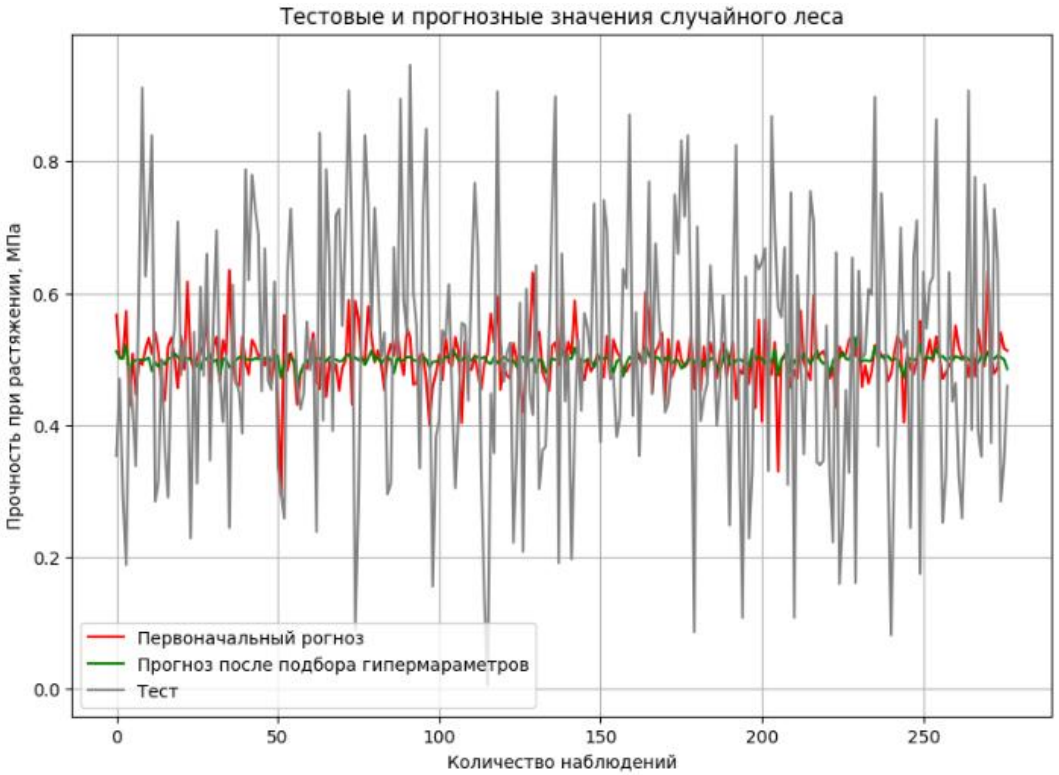

Рисунок 21 Результаты прогнозирования прочности при растяжении методом градиентного бустинга

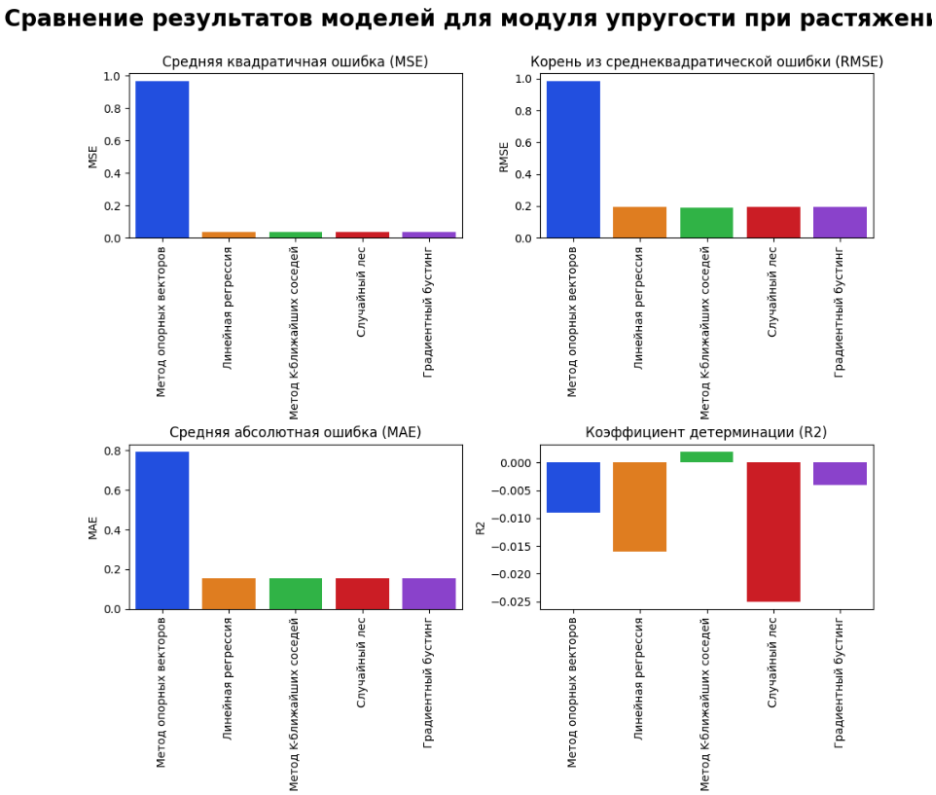

Рисунок 22 Сравнение метрик моделей для прогнозирования модуля упругости при растяжении

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

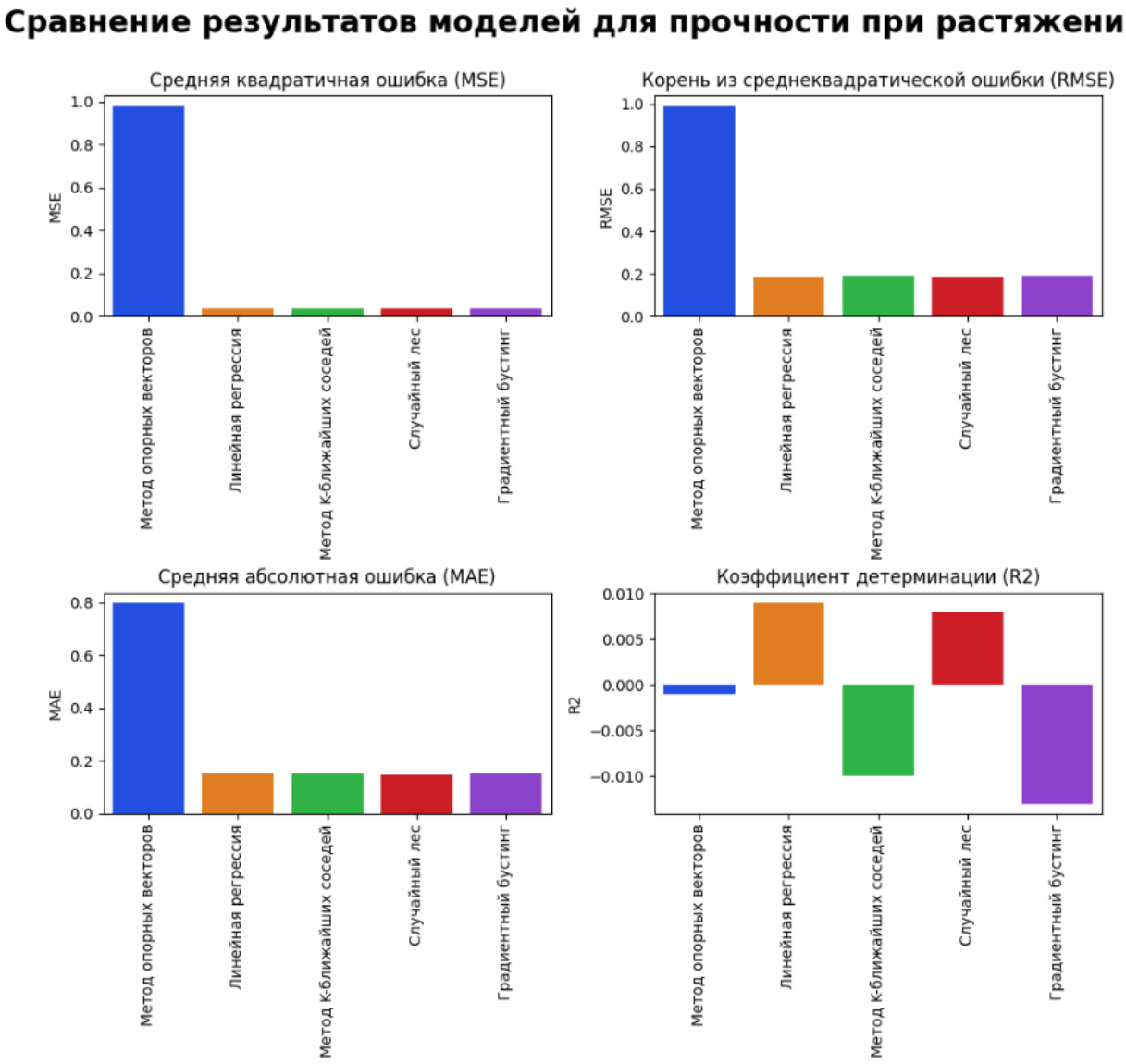

Рисунок 23 Сравнение метрик моделей для прогнозирования прочности при растяжении

Как видно из приведенных рисунков, ни одна из рассмотренных моделей не дала удовлетворительного результата.

4 Разработка и обучение нейронной сети для прогнозирования Соотношения матрица-наполнитель.

Целевой признак – Соотношение матрица-наполнитель.

На вход моделям будем подавать 12 признаков:

- Плотность, кг/м3;

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

- Модуль упругости, Гпа;
- Количество отвердителя, м.%;
- Содержание эпоксидных групп, %_2;
- Температура вспышки, С_2;
- Поверхностная плотность, г/м2;
- Модуль упругости при растяжении, ГПа
- Прочность при растяжении;
- Потребление смолы, г/м2;
- Угол нашивки, град;
- Шаг нашивки;
- Плотность нашивки.

Нейронную сеть строим с помощью класса Sequential библиотеки keras (рисунок 24).

Данная нейронная сеть имеет следующие параметры:

– входной слой

– первый полносвязный слой с 8 нейронами и функцией активации tanh;

– второй полносвязный слой с 8 нейронами и функцией активации tanh;

– один Dropout слой;

– выходной полносвязный слой с 1 нейроном и линейной функцией активации;

– loss-функция: среднеквадратичная ошибка (mean_squared_error).

На рисунках 24, 25 видно, что нейросеть также не показала удовлетворительных результатов

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

```python
def base_model():
    model = Sequential()
    model.add(Dense(8, input_dim=12, activation='tanh')) # скрытый полносвязный слой 1 input_dim=12
    model.add(Dense(8, activation='tanh')) # скрытый полносвязный слой 2
    model.add(Dropout(0.5))
    model.add(Dense(1, activation='linear')) # выходной слой

    model.compile(loss='mean_squared_error', optimizer='sgd')
    return model

#Создаем НС и обучаем её
ns = base_model()
history = ns.fit(X_train_ns, y_train_ns,
                 epochs=400,
                 verbose=0, validation_data=(X_test_ns, y_test_ns))

#Предсказываем значения
y_pred_ns = ns.predict(X_test_ns)

#Печатаем метрики и график
print_metrics(y_test_ns, y_pred_ns)
```

```
[75] ✓ 49.6s
9/9 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step
Среднеквадратическая ошибка MSE: 0.037
Средняя абсолютная ошибка MAE: 0.156
Корень из среднеквадратической ошибки RMSE: 0.193
Коэффициент детерминации R2: -0.005
Точность модели (%) 68.785
```

Рисунок 24 Нейронная сеть

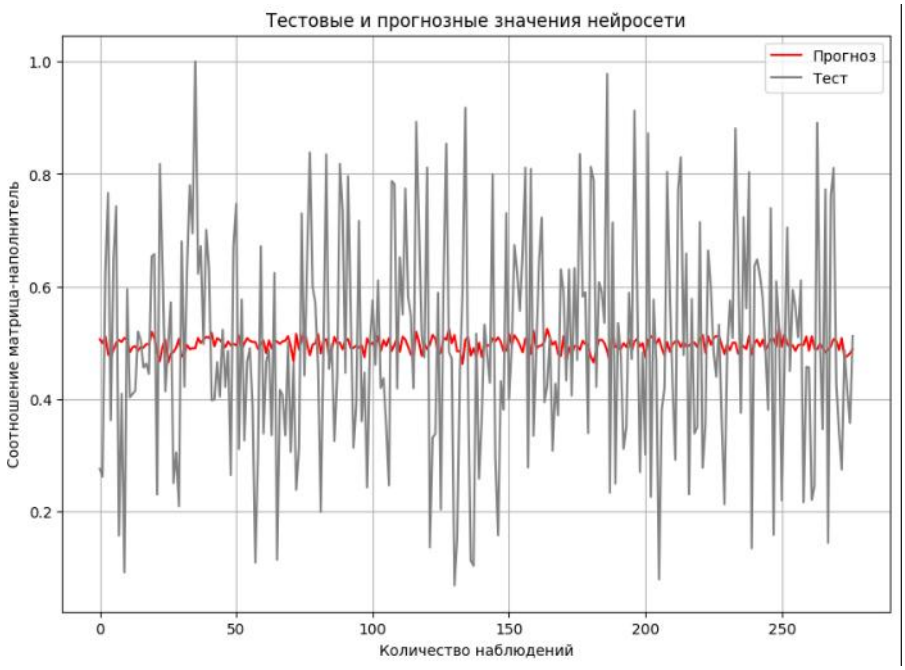

Рисунок 45 Сравнительные графики тестовых и прогнозируемых значений нейросетью

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

5 Консольное приложение для прогнозирования соотношения матрица-наполнитель

Архитектура консольного приложения для прогнозирования соотношения матрица-наполнитель приведена на рисунке 27

```python
# Создадим функцию для ввода данных
def input_variable():
    x1 = float(input('Плотность, кг/м3: '))
    x2 = float(input('Модуль упругости, ГПа: '))
    x3 = float(input('Количество отвердителя, м.%: '))
    x4 = float(input('Содержание эпоксидных групп,%_2: '))
    x5 = float(input('Температура вспышки, С_2: '))
    x6 = float(input('Поверхностная плотность, г/м2: '))
    x7 = float(input('Модуль упругости при растяжении, ГПа: '))
    x8 = float(input('Прочность при растяжении, МПа: '))
    x9 = float(input('Потребление смолы, г/м2: '))
    x10 = float(input('Угол нашивки: '))
    x11 = float(input('Шаг нашивки: '))
    x12 = float(input('Плотность нашивки: '))
    return x1, x2, x3, x4, x5, x6, x7, x8, x9, x10, x11, x12
```

✓ 0.0s

```python
# Создадим функцию для вызова приложения
def app_model():
    # Загружаем модель и масштабаторы
    nn_model = load_model('../model_dir/my_model.keras')
    scaler_x = load('../model_dir/minmax_scl_x.pkl')
    scaler_y = load('../model_dir/minmax_scl_y.pkl')

    print('Приложение прогнозирует соотношение "матрица-наполнитель"')
    for i in range(110):
        try:
            print('Введите "1" для прогноза, "2" для выхода')
            check = input()

            if check == '1':
                print('Введите данные для прогноза')
                X = input_variable()
                X =  scaler_x.transform(np.array(X).reshape(1,-1))
                prediction = nn_model.predict(X)
                output = scaler_y.inverse_transform(prediction)
                print('Прогнозное значение соотношения "матрица-наполнитель": ')
                print(output[0][0])

            elif check == '2':
                break
            else:
                print('Повторите выбор')

        except Exception as e:
            print(e)
            print('Введены некорректные данные. Пожалуйста, повторите операцию')
app_model()
```

✓ 5m 2.8s

```
Приложение прогнозирует соотношение "матрица-наполнитель"
Введите "1" для прогноза, "2" для выхода
```

Рисунок 27 – Консольное приложение для прогнозирования соотношения матрица-наполнитель

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

Заключение

Рассмотренные модели машинного обучения: метод опорных векторов, линейная регрессия, метод К-ближайших соседей, случайный лес, градиентный бустинг, нейронная сеть показали неудовлетворительные результаты при прогнозировании целевых переменных даже с учетом подбора гиперпараметров.

Для достижения целей работы необходимо провести более обширное исследование, включающее:

- привлечение большего количества моделей машинного обучения;
- осуществление более тщательный подбора гиперпараметров для каждой модели
- экспертная оценка фактического влиянии исследуемых параметров на целевые переменные, и на основании нее избавиться от малозначительных переменных, или провести дополнительные эксперименты.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 182-83-85, do@bmstu.ru
do.bmstu.ru

Библиографический список

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

2) Документация по библиотеке numpy: – Режим доступа: https://numpy.org/doc/1.22/user/index.html#user.

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

4) Документация по библиотеке matplotlib: – Режим доступа: https://matplotlib.org/stable/users/index.html.

5) Документация по библиотеке seaborn: – Режим доступа: https://seaborn.pydata.org/tutorial.html.

6) Документация по библиотеке sklearn: – Режим доступа: https://scikit-learn.org/stable/user_guide.html.

7) Документация по библиотеке keras: – Режим доступа: https://keras.io/api/.

8) Loginom Вики. Алгоритмы: – Режим доступа: https://wiki.loginom.ru/algorithms.html.