

**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

Ефремов Ярослав Владиславович

Москва, 2024

Содержание

Содержание	2
Введение	3
1. Аналитическая часть	5
1.1. Постановка задачи.....	5
1.2. Разведочный анализ данных	8
1.3. Описание используемых методов.....	15
2. Практическая часть	18
2.1. Предобработка данных	18
2.2. Разработка и обучение модели	20
2.3. Написать нейронную сеть, которая будет рекомендовать	22
Разработка приложения	25
2.4. Создание удалённого репозитория и загрузка	28
Заключение.....	29
2.5. Список используемой литературы и веб ресурсы.....	31

Введение

Композиционные материалы — это материалы, состоящие из двух или более компонентов, нерастворимых друг с другом, с чётко обозначенной границей раздела и сильным взаимодействием по всей зоне контакта. Одним из компонентов композитных материалов является непрерывная фаза, он называется матрица, в которой нерастворимые материалы помещаются в другую природу, называемую арматурой или наполнителем.

Внедрение композиционных материалов обусловлено стремлением использовать их преимущества по сравнению с традиционно используемыми металлами и сплавами. Примеры композита – железобетон (сочетание стали арматуры и камня бетона), древесноволокнистая плита ДВП (сочетание древесной основы – щепы и полимерного связующего).

Базальт - магматическая вулканическая порода. Это самая распространённая порода на поверхности Земли и на других планетах Солнечной системы. Базальты образуются путём затвердевания силикатного магматического расплава. Большая часть базальтов образуется на срединно-океанических хребтах и образует океаническую кору. Активно развивается использование композитных материалов на основе базальта.

Базальтопластик - современный композитный материал на основе базальтовых волокон и органического связующего вещества. В настоящее время базальтопластик успешно конкурирует с металлическими изделиями, превосходя их по коррозионной, щелочной, кислотоустойчивости и некоторым другим свойствам. Целью данной работы является прогнозирование конечных свойств новых материалов на основе базальтопластика (композиционных материалов).

Расширение разнообразия материалов, используемых при проектировании нового композиционного материала, увеличивает необходимость определения свойств нового композита при минимальных финансовых затратах. Для решения

этой проблемы обычно используются два способа: физические тесты образцов материалов или оценка свойств, в том числе на основе физико-математических моделей. Традиционно разработка композитных материалов является долгосрочным процессом, так как из свойств отдельных компонентов невозможно рассчитать конечные свойства композита.

Возможно получить композиты с уникальными эксплуатационными свойствами. Этим обусловлено широкое применение композиционных материалов в различных областях техники. Композиционные материалы используются:

- в авиационной, ракетной и космической технике;
- в металлургии;
- в горнорудной промышленности;
- в химической промышленности;
- в автомобильной промышленности;
- сельскохозяйственном машиностроении;
- в электротехнической промышленности;
- в ядерной технике;
- в машиностроительной отрасли;
- в сварочной технике;
- судостроительной промышленности;
- в медицинской промышленности;
- в строительстве;
- в бытовой технике.

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

Стоимость производства композитного материала высока. Зная характеристики компонентов, невозможно рассчитать свойства композита. Значит для получения заданных свойств требуется большое количество испытаний различных

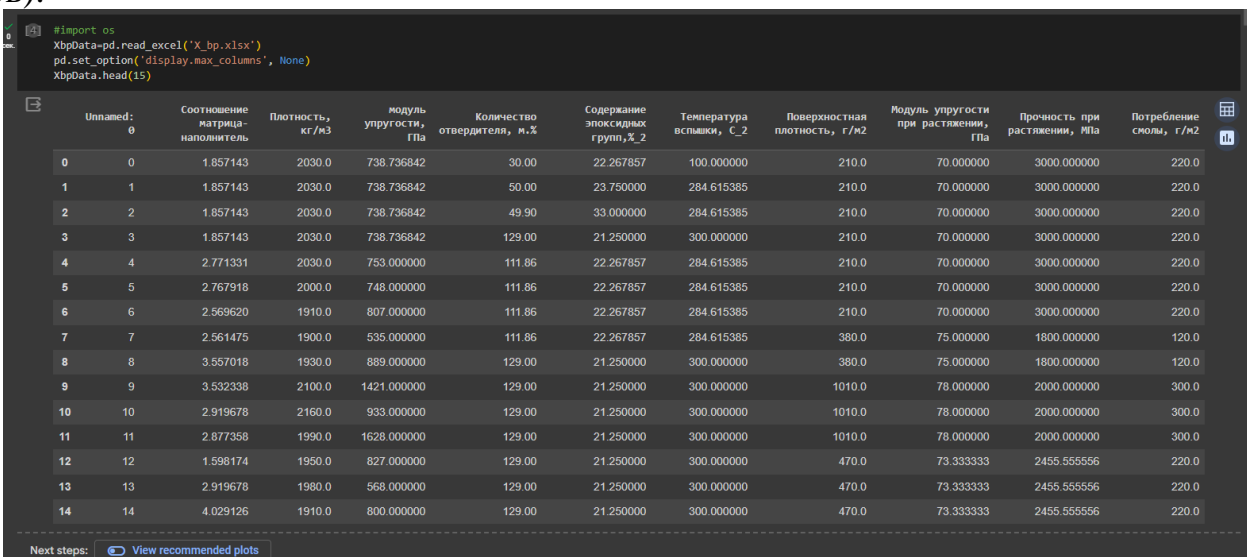
комбинаций. Сократить время и затраты на создание определенного материала могла бы помочь система поддержки производственных решений, построенная на принципах машинного обучения.

Суть прогнозирования заключается в моделировании репрезентативного элемента композитного объёма на основе данных о свойствах входящих компонентов (связующего и армирующего компонента). В процессе исследовательской работы были разработаны несколько моделей, способные с высокой вероятностью прогнозировать модули упругости при растяжении и прочности при растяжении, а также были созданы нейронная сеть, которая предлагает соотношение «матрицы - наполнитель». Также разработано приложение на библиотеке Aiogram 3, специализированная на чат-ботах Telegram.

1. Аналитическая часть

1.1. Постановка задачи

Для исследовательской работы были даны 2 файла: X_bp.xlsx (с данными о параметрах базальтопластика, состоящий из 1024 строки и 11 столбцов) и X_nipr.xlsx (данными нашивок углепластика, состоящий из 1041 строки и 4 столбцов).



```
#import os
XbpData=pd.read_excel('X_bp.xlsx')
pd.set_option('display.max_columns', None)
XbpData.head(15)
```

Unnamed: 0	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
0	0	1.857143	2030.0	738.736842	30.00	22.267857	100.000000	210.0	70.000000	3000.000000
1	1	1.857143	2030.0	738.736842	50.00	23.750000	284.615385	210.0	70.000000	3000.000000
2	2	1.857143	2030.0	738.736842	49.90	33.000000	284.615385	210.0	70.000000	3000.000000
3	3	1.857143	2030.0	738.736842	129.00	21.250000	300.000000	210.0	70.000000	3000.000000
4	4	2.771331	2030.0	753.000000	111.86	22.267857	284.615385	210.0	70.000000	3000.000000
5	5	2.767918	2000.0	748.000000	111.86	22.267857	284.615385	210.0	70.000000	3000.000000
6	6	2.569620	1910.0	807.000000	111.86	22.267857	284.615385	210.0	70.000000	3000.000000
7	7	2.561475	1900.0	535.000000	111.86	22.267857	284.615385	380.0	75.000000	1800.000000
8	8	3.557018	1930.0	889.000000	129.00	21.250000	300.000000	380.0	75.000000	1800.000000
9	9	3.532338	2100.0	1421.000000	129.00	21.250000	300.000000	1010.0	78.000000	2000.000000
10	10	2.919678	2160.0	933.000000	129.00	21.250000	300.000000	1010.0	78.000000	2000.000000
11	11	2.877358	1990.0	1628.000000	129.00	21.250000	300.000000	1010.0	78.000000	2000.000000
12	12	1.598174	1950.0	827.000000	129.00	21.250000	300.000000	470.0	73.333333	2455.555556
13	13	2.919678	1980.0	568.000000	129.00	21.250000	300.000000	470.0	73.333333	2455.555556
14	14	4.029126	1910.0	800.000000	129.00	21.250000	300.000000	470.0	73.333333	2455.555556

Рисунок 1 – X_bp.xlsx

Цель работы разработать модели для прогноза модуля упругости при растяжении, прочности при растяжении и соотношения «матрица-наполнитель». Для этого нужно объединить 2 файла. Часть информации (17 строк таблицы способов компоновки композитов) не имеют соответствующих строк в таблице соотношений и свойств используемых компонентов композитов, поэтому были удалены.

Описание признаков объединенного датасета приведено в таблице 1. Все признаки имеют тип float64, то есть вещественный. Пропусков в данных нет. Все признаки, кроме «Угол нашивки», являются непрерывными, количественными. «Угол нашивки» принимает только два значения и будет рассматриваться как категориальный признак.

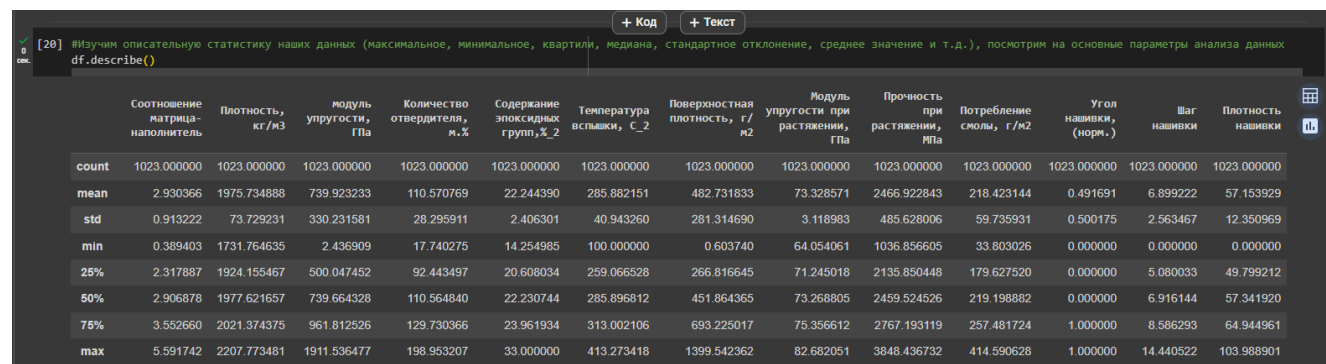
Таблица 1 — Описание признаков датасета

Название	Файл	Тип данных	Непустых значений	Уникальных значений
Соотношение матрица-наполнитель	X_bp	float64	1023	1014
Плотность, кг/м ³	X_bp	float64	1023	1013
модуль упругости, ГПа	X_bp	float64	1023	1020
Количество отвердителя, м.%	X_bp	float64	1023	1005
Содержание эпоксидных групп,%_2	X_bp	float64	1023	1004
Температура вспышки, С_2	X_bp	float64	1023	1003
Поверхностная плотность, г/м ²	X_bp	float64	1023	1004

Таблица 1 – Продолжение

Модуль упругости при растяжении, ГПа	X_bp	float64	1023	1004
Прочность при растяжении, МПа	X_bp	float64	1023	1004
Потребление смолы, г/м2	X_bp	float64	1023	1003
Угол нашивки, град	X_nup	float64	1023	2
Шаг нашивки	X_nup	float64	1023	989
Плотность нашивки	X_nup	float64	1023	988

Методом `df.describe()` смотрим описательную статистику (Рисунок 3). Метод показывает количество значений, среднее значение, стандартное отклонение, минимальное значение, 25-50-75% перцентили и максимальное значение.



[20] #Изучим описательную статистику наших данных (максимальное, минимальное, квартили, медиана, стандартное отклонение, среднее значение и т.д.), посмотрим на основные параметры анализа данных

```
df.describe()
```

	Соотношение матрица-наполнитель	Плотность, кг/м3	модуль упругости, ГПа	Количество отвердителя, м.%	Содержание эпоксидных групп, %_2	Температура вспышки, C_2	Поверхностная плотность, г/ м2	Модуль упругости при растяжении, ГПа	Прочность при растяжении, МПа	Потребление смолы, г/м2	Угол нашивки, (норм.)	Шаг нашивки	Плотность нашивки
count	1023.000000	1023.000000	1023.000000	1023.000000	1023.000000	1023.000000	1023.000000	1023.000000	1023.000000	1023.000000	1023.000000	1023.000000	1023.000000
mean	2.930366	1975.734888	739.923233	110.570769	22.244390	285.882151	482.731833	73.328571	2466.922843	218.423144	0.491691	6.899222	57.153929
std	0.913222	73.729231	330.231581	28.295911	2.406301	40.943260	281.314690	3.118983	485.628006	59.735931	0.500175	2.563467	12.350969
min	0.389403	1731.764635	2.436909	17.740275	14.254985	100.000000	0.603740	64.054061	1036.856605	33.803026	0.000000	0.000000	0.000000
25%	2.317887	1924.155467	500.047452	92.443497	20.608034	259.066528	266.816645	71.245018	2135.850448	179.627520	0.000000	5.080033	49.799212
50%	2.906878	1977.621657	739.664328	110.564840	22.230744	285.896812	451.864365	73.268805	2459.524526	219.198882	0.000000	6.916144	57.341920
75%	3.552660	2021.374375	961.812526	129.730366	23.961934	313.002106	693.225017	75.356612	2767.193119	257.481724	1.000000	8.586293	64.944961
max	5.591742	2207.773481	1911.536477	198.953207	33.000000	413.273418	1399.542362	82.682051	3848.436732	414.590628	1.000000	14.440522	103.988901

Рисунок 2 – Метод `df.describe()`

«Угол нашивки» принимает только два значения и будет рассматриваться как категориальный признак, мы заменяем значение 0 и 90 градусов, на 0 и 1, это приведено на рисунке 3.

```
[17] # Приведем столбец "Угол нашивки" к значениям 0 и 1
      df['Угол нашивки, град'].replace({0: 0, 90: 1}, inplace=True)
      df['Угол нашивки, град']
```

0	0
1	0
2	0
3	0
4	0
..	
1018	1
1019	1
1020	1
1021	1
1022	1

Name: Угол нашивки, град, Length: 1023, dtype: int64

Рисунок 3 – Угол нашивки

1.2. Разведочный анализ данных

Прежде чем передать данные в работу моделей машинного обучения, необходимо обработать и очистить их. Очевидно, что «грязные» и необработанные данные могут содержать искажения и пропущенные значения – это ненадёжно, поскольку способно привести к крайне неверным результатам по итогам моделирования. Но безосновательно удалять что-либо тоже неправильно. Именно поэтому сначала набор данных надо изучить.

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.

Затем провести разведочный анализ данных, нарисовать гистограммы распределения каждой из переменной, диаграммы «ящик с усами», попарные графики рассеяния точек, приведены на рисунках 4 – 9.

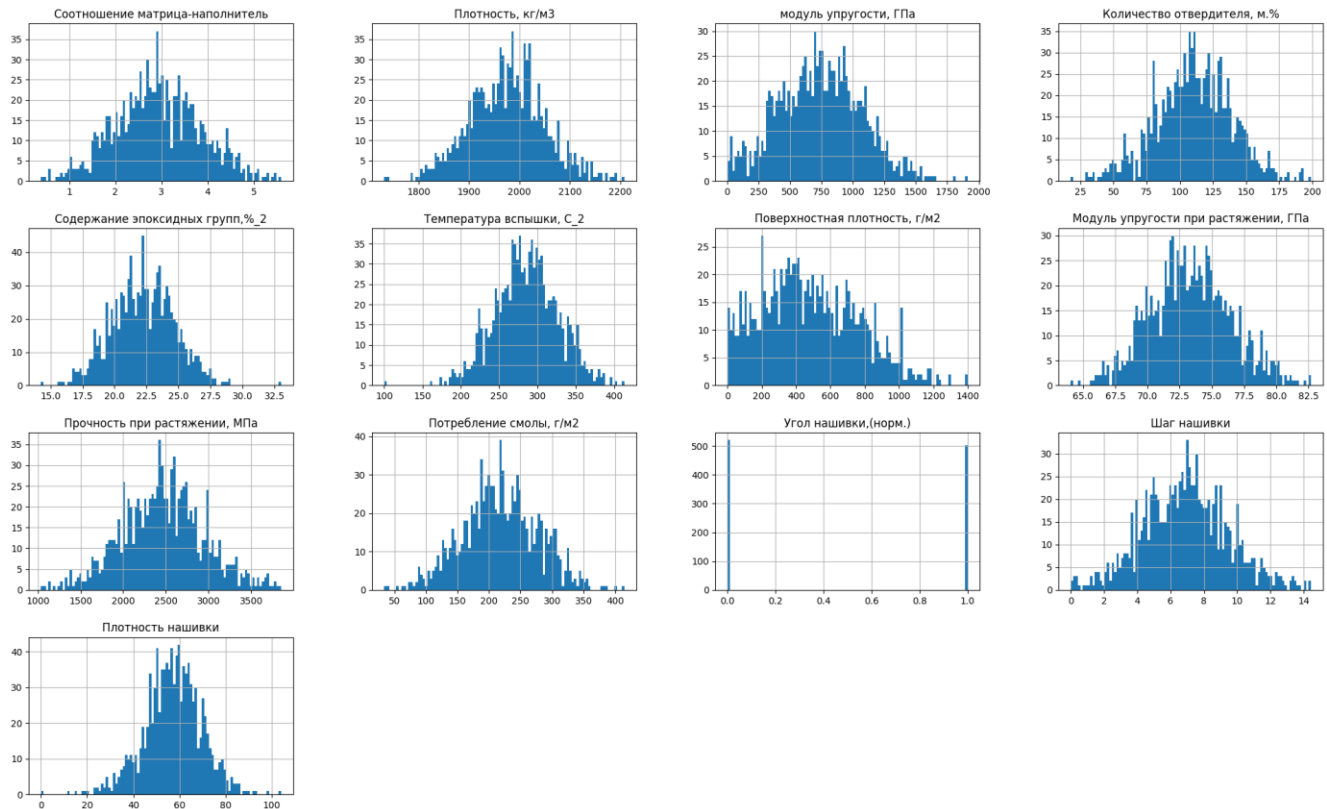


Рисунок 1 – Гистограммы распределения

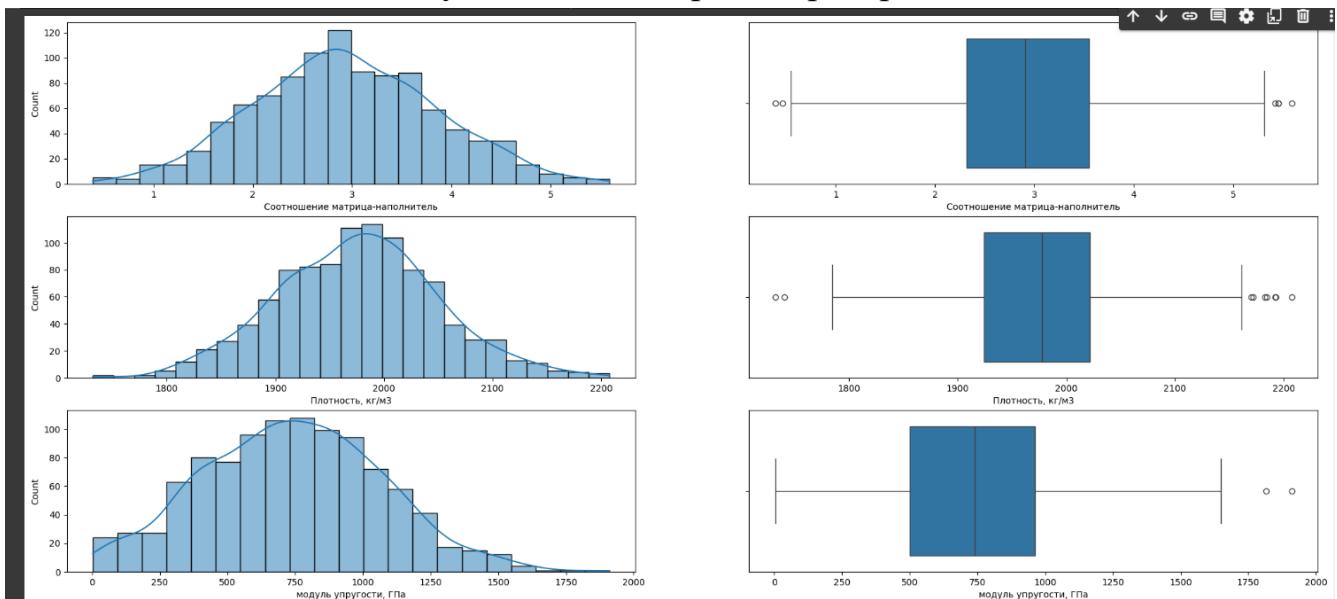


Рисунок 5 – Диаграмма "ящик с усами" в объединённом датасете

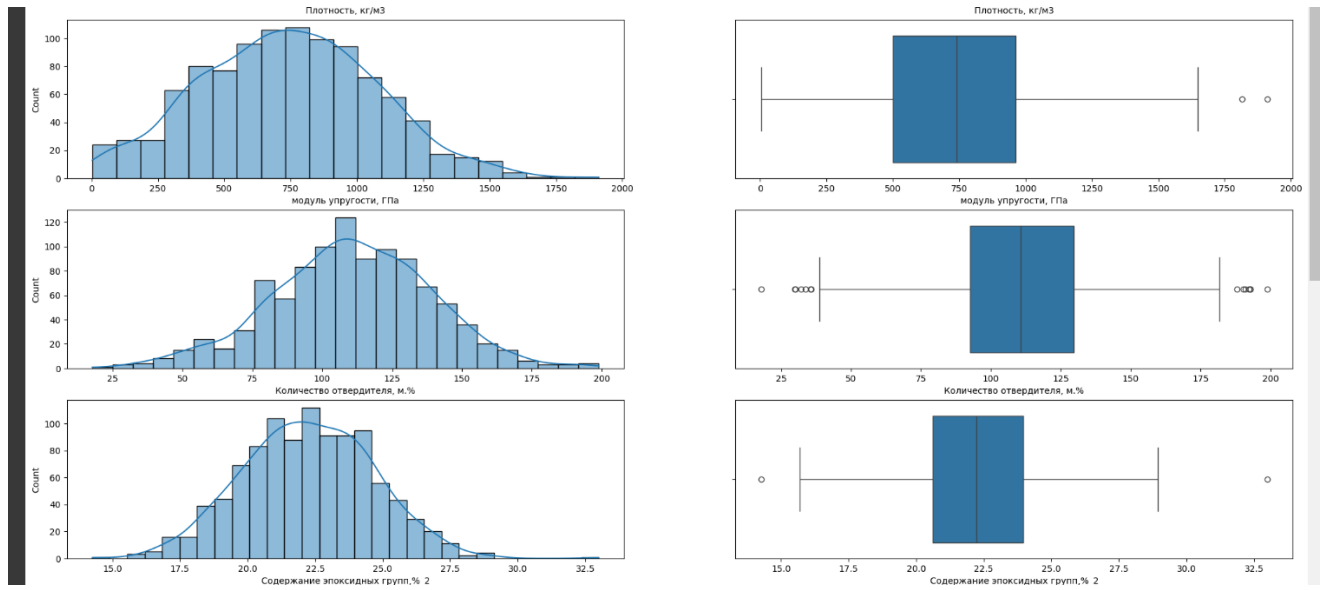


Рисунок 6 – Диаграмма "ящик с усами" в объединённом датасете

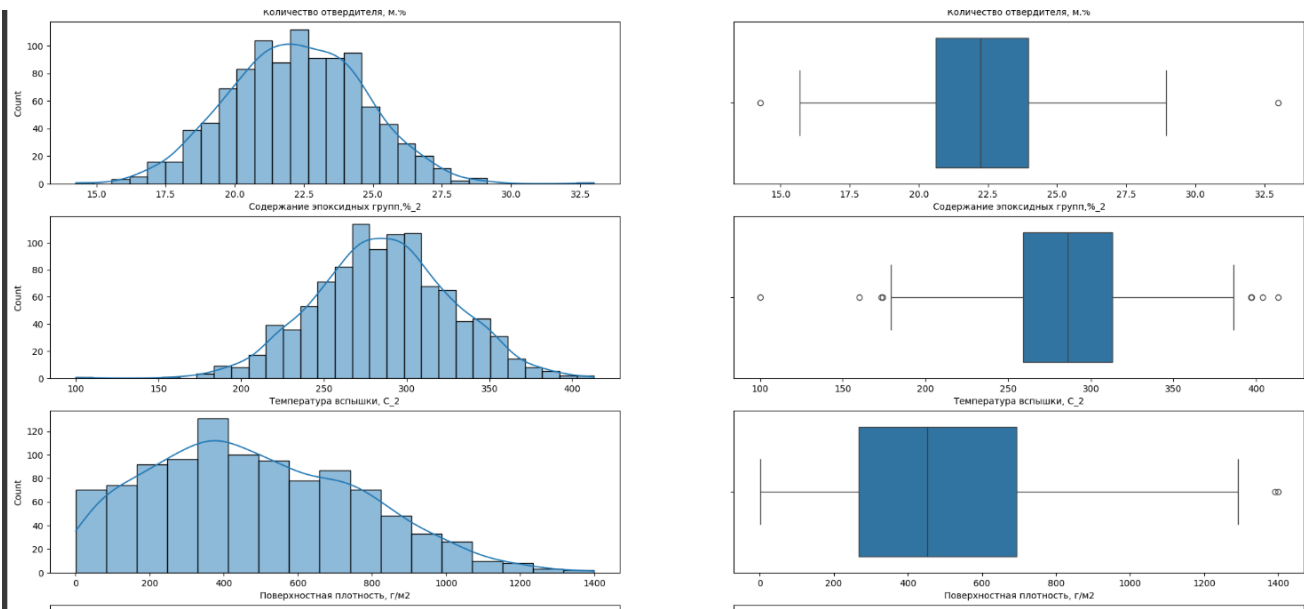


Рисунок 7 – Диаграмма "ящик с усами" в объединённом датасете

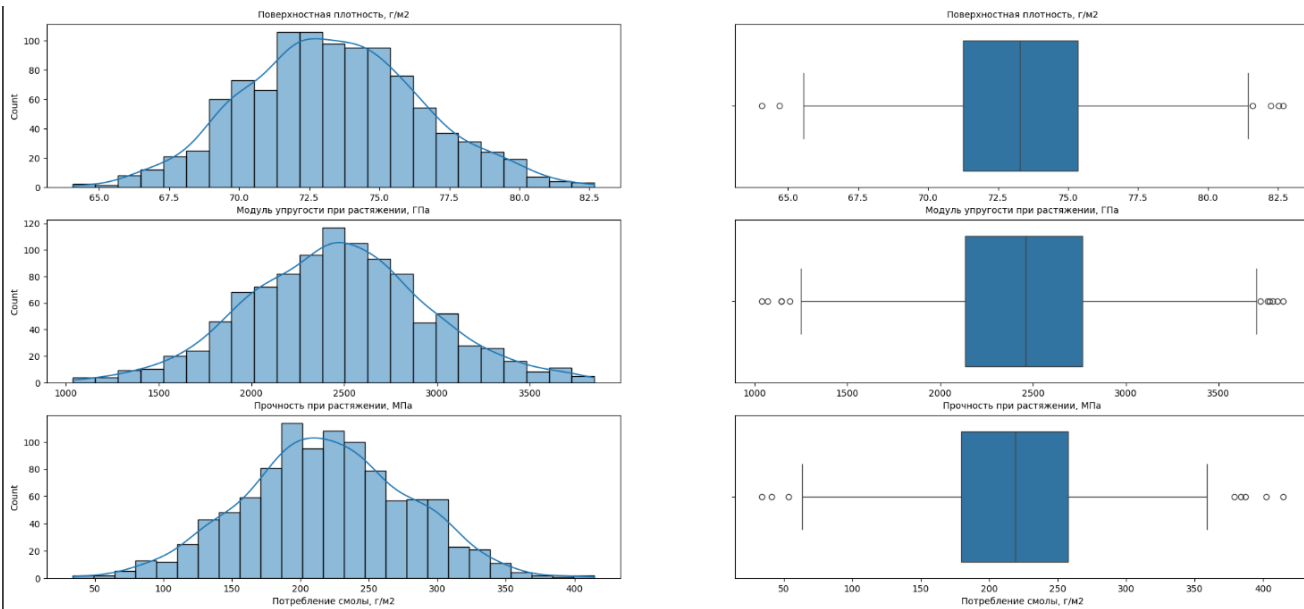


Рисунок 8 – Диаграмма "ящик с усами" в объединённом датасете

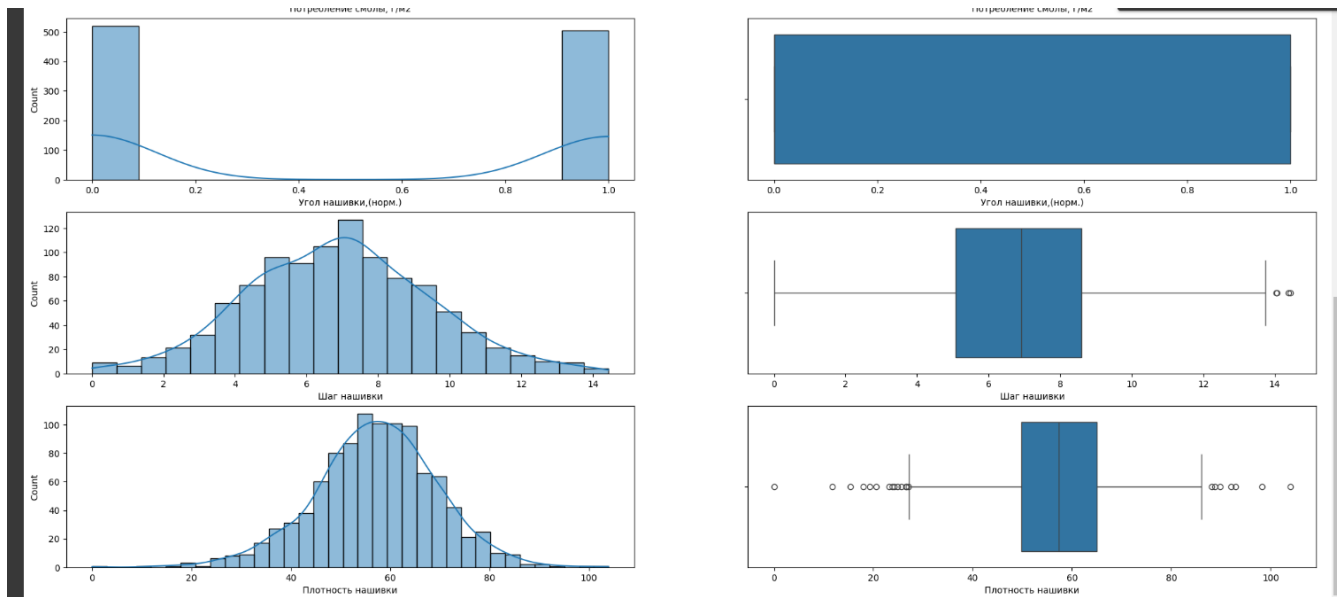


Рисунок 9 – Диаграмма "ящик с усами" в объединённом датасете

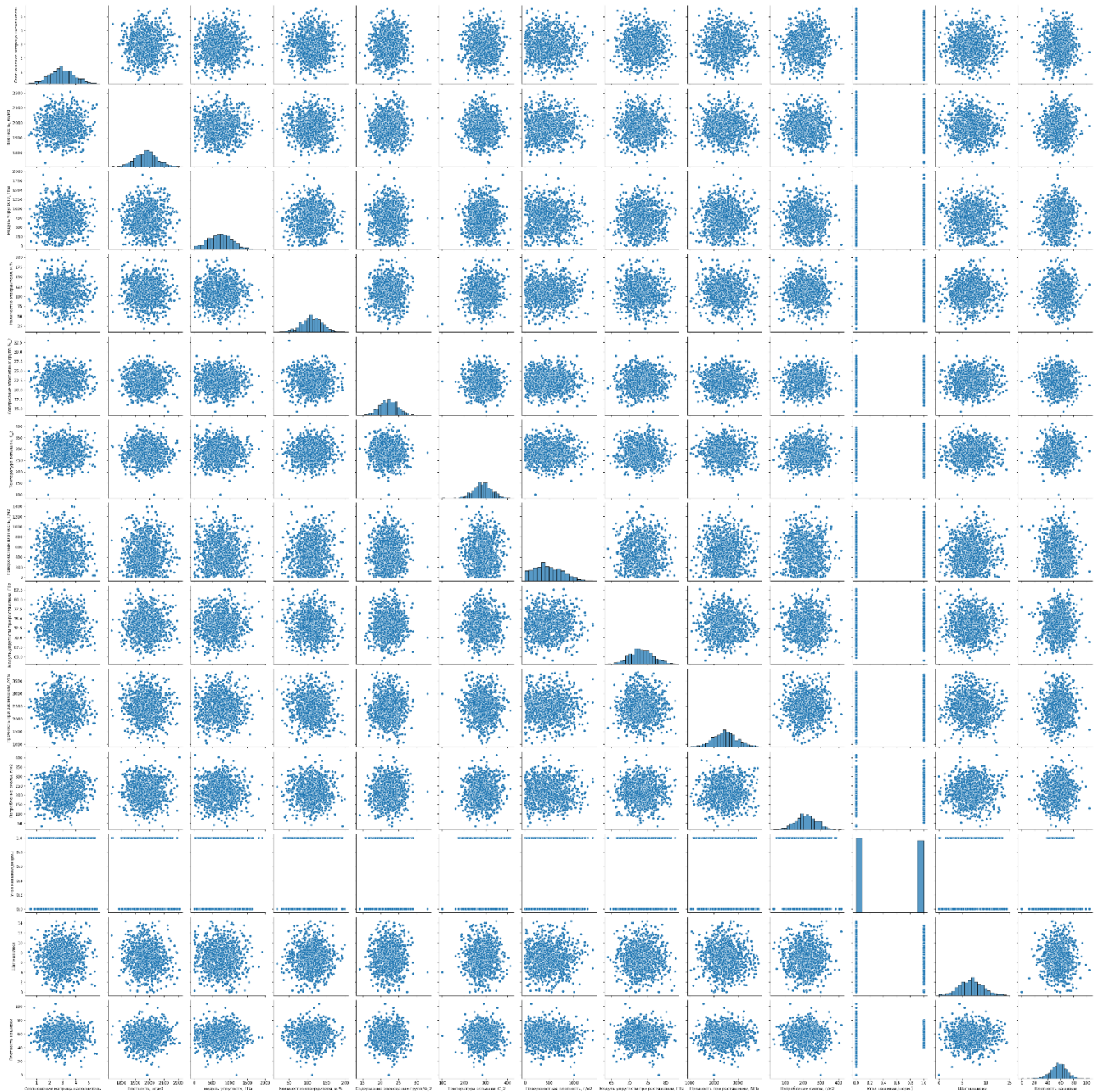


Рисунок 10 – Попарные графики рассеяния точек

По графикам рассеяния мы видим, что некоторые точки отстоят далеко от общего облака. Так визуально выглядят выбросы — аномальные, некорректные значения данных, выходящие за пределы допустимых значений признака.

Стоит обратить внимание, что взаимосвязь корреляция между признаками нет, что данные для такой практической задачи требуется тщательный анализ, чтобы найти корреляцию между признаками.

Есть следующие методы выявления выбросов для признаков с нормальным распределением:

- метод 3-х сигм;
- метод межквартильных расстояний.

Применив эти методы на нашем датасете, было найдено:

- методом 3-х сигм — 24 выброса;
- методом межквартильных расстояний — 93 выброса.

Пример выбросов на гистограмме распределения и диаграмме «ящик с усами» приведен на рисунке 11.

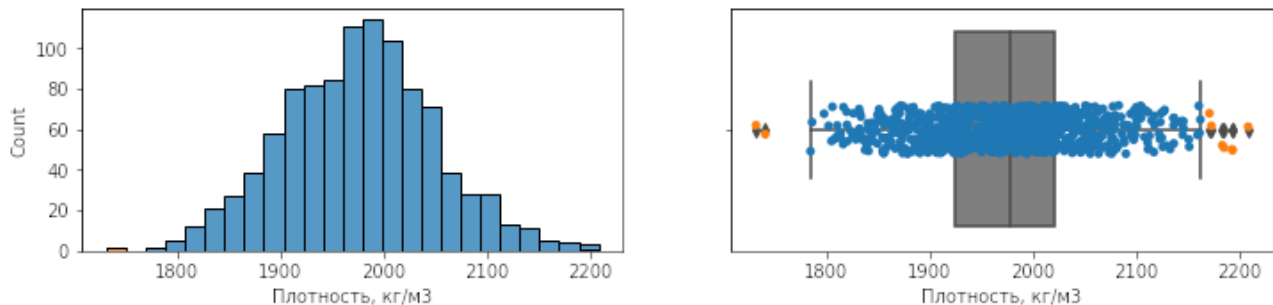


Рисунок 11 – Пример выбросов

Поскольку известно, что датасет очищен от явного шума, следует применить метод 3-х сигм как более деликатный, чтобы не потерять значимые данные. Значения, определенные как выбросы, удаляем. После этого осталось в датасете осталось 1000 строк и 13 признаков-переменных.

В задании целевыми переменными указаны:

- модуль упругости при растяжении, ГПа;
- прочность при растяжении, МПа;
- соотношение матрица-наполнитель.

Для каждой колонки получить среднее, медианное значение, провести анализ и исключение выбросов, проверить наличие пропусков; пред обработать данные: 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 «прочность при растяжении, МПа». Оценить точность модели на тренировочном и тестовом датасете. Создать репозиторий в GitHub и разместить код исследования. Оформить файл README.

1.3. Описание используемых методов

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

- случайный лес;
- линейная регрессия;
- К-ближайших соседей;
- дерево решений;

Случайный лес (RandomForest) — это множество решающих деревьев. Универсальный алгоритм машинного обучения с учителем, представитель ансамблевых методов. Если точность дерева решений оказалась 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.

Линейная регрессия (Linear regression) — это алгоритм машинного обучения, основанный на контролируемом обучении, рассматривающий зависимость между одной входной и выходными переменными. Это один из самых простых и эффективных инструментов статистического моделирования. Она определяет зависимость переменных с помощью линии наилучшего соответствия. Модель регрессии создаёт несколько метрик. 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 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.

Метод ближайших соседей - K-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.

Дерево принятия решений (DecisionTreeRegressor) – метод автоматического анализа больших массивов данных. Это инструмент принятия решений, в котором используется древовидная структура, подобная блок-схеме, или модель решений и всех их возможных результатов, включая результаты, затраты и полезность. Дерево принятия решений - эффективный инструмент интеллектуального анализа данных и предсказательной аналитики. Алгоритм дерева решений подпадает под категорию контролируемых алгоритмов обучения. Он работает как для непрерывных, так и для категориальных выходных переменных. Правила генерируются за счёт обобщения множества отдельных наблюдений (обучающих примеров), описывающих предметную область. Регрессия дерева решений отслеживает особенности объекта и обучает модель в структуре дерева прогнозированию данных в будущем для получения значимого непрерывного вывода. Дерево решений один из вариантов решения регрессионной задачи, в случае если зависимость в данных не имеет очевидной корреляции.

Достоинства метода: помогают визуализировать процесс принятия решения и сделать правильный выбор в ситуациях, когда результаты одного решения

влияют на результаты следующих решений; создаются по понятным правилам; просты в применении и интерпретации; заполняют пропуски в данных наиболее вероятным решением; работают с разными переменными; выделяют наиболее важные поля для прогнозирования;

Недостатки метода: ошибаются при классификации с большим количеством классов и небольшой обучающей выборкой; имеют нестабильный процесс (изменение в одном узле может привести к построению совсем другого дерева); имеет затратные вычисления; необходимо обращать внимание на размер; ограниченное число вариантов решения проблемы.

2. Практическая часть

2.1. Предобработка данных

Так как значения не коррелируются с друг другом, то нужно исследовать предобработку данных и получить желанную корреляцию между данными.

В анализе участвуют:

- `MinMaxScaler()`
- `Normalizer()`
- `RobustScaler()`

`StandardScaler` не будем использовать, так как некоторые данные нет нормального распределения, так что не будем его использовать, будем использовать другие методы предпроцессинга.

По условиям задания нормализуем значения. Результатов исследование представление графики корреляции.

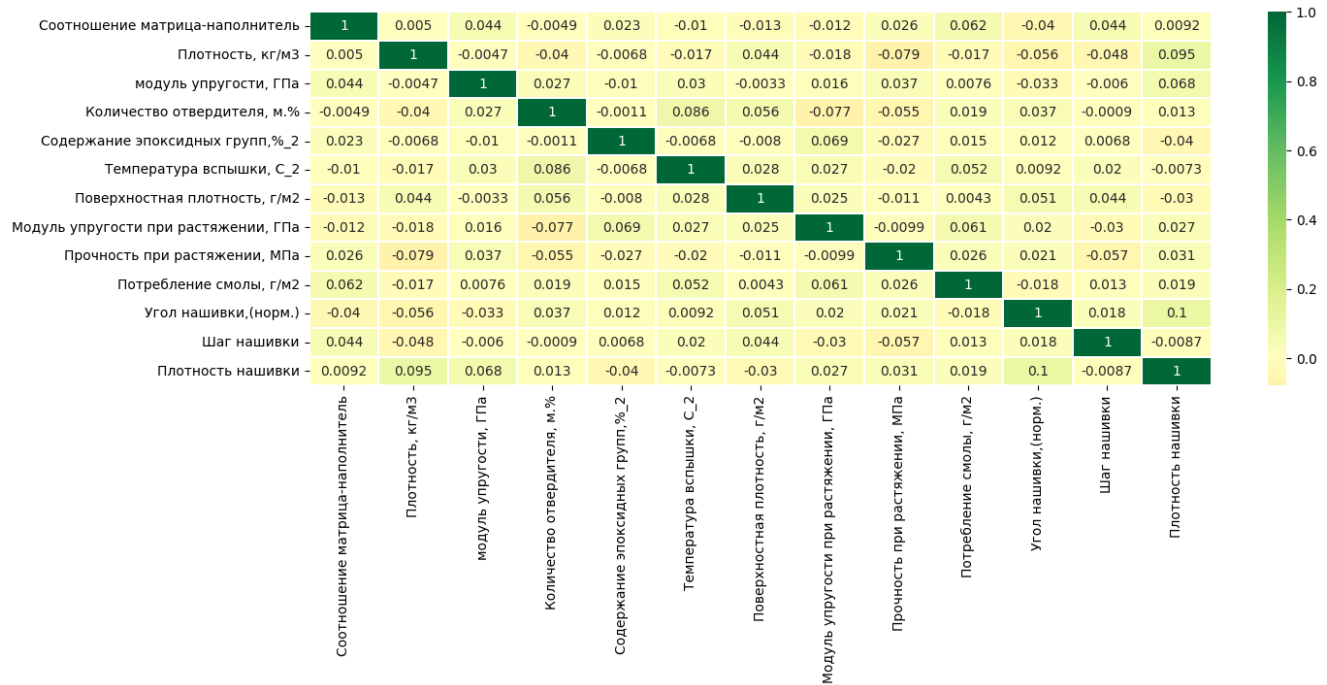


Рисунок 12 – Визуализированные данные до и после нормализации



Рисунок 13 – Визуализированные данные до и после нормализации

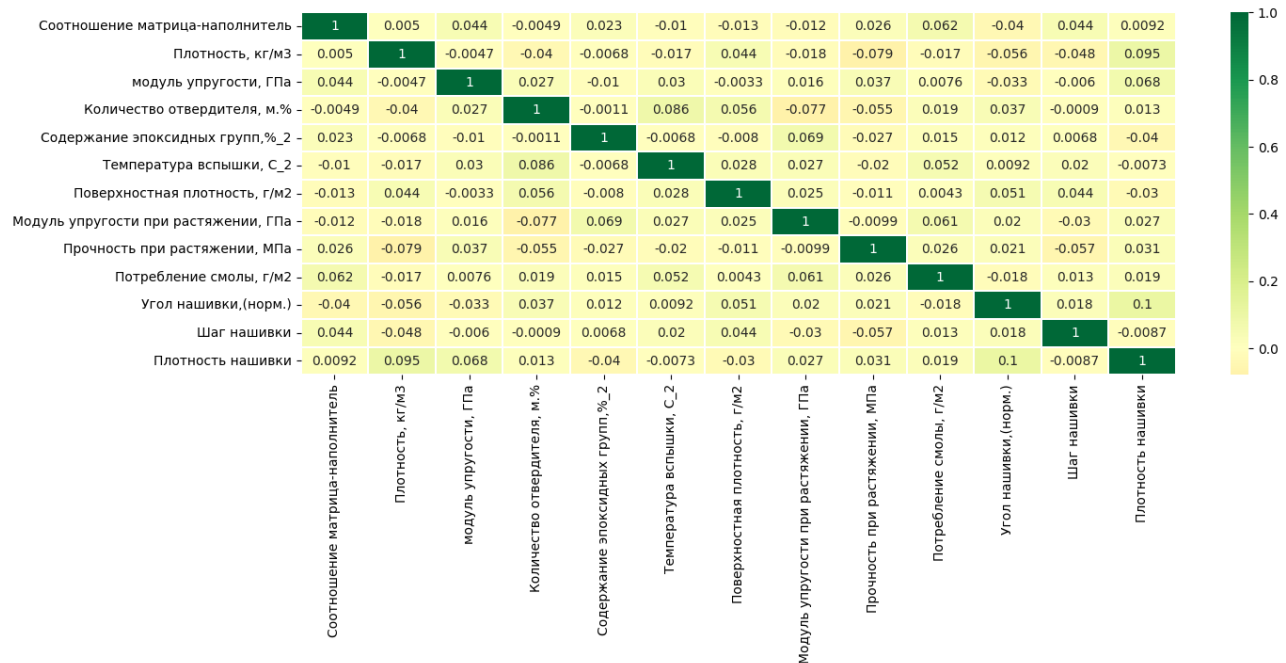


Рисунок 14 – Визуализированные данные до и после нормализации

Предпроцессинг Normalizer() дает нам зависимость между данными и ее корреляцию, плюс можно увидеть, что модели машинного обучения используем регрессию.

2.2. Разработка и обучение модели

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

Порядок разработки модели для каждого параметра и для каждого выбранного метода можно разделить на следующие этапы: разделение нормализованных данных на обучающую и тестовую выборки (в соотношении 70 на 30%); проверка моделей при стандартных значениях; сравнение с результатами модели, выдающей среднее значение; создание графика; сравнение моделей по метрике MAE; поиск сетки гиперпараметров, по которым будет происходить оптимизация модели. В качестве параметра оценки выбран коэффициент детерминации (R2); оптимизация

подбора гиперпараметров модели с помощью выбора по сетке и перекрёстной проверки; подстановка оптимальных гиперпараметров в модель и обучение модели на тренировочных данных; оценка полученных данных; сравнение со стандартными значениями.

Модель после настройки гиперпараметров показала результат немного лучше. Однако, ниже, чем базовая модель. Прочность при растяжении и модуль упругости не имеет линейной зависимости. Все использованные модели не справились с задачей. Результат неудовлетворительный. Свойства композитных материалов в первую очередь зависят от используемых материалов.

Таблица 2 – Результаты построения и обучения моделей

	Model	MAE	R2 score
Прочность при растяжении	RandomForestRegressor_pr	0.008056	0.963
Прочность при растяжении	KNeighborsRegressor_pr	0.008650	0.962
Прочность при растяжении	LinearRegression_pr	0.011843	0.950
Прочность при растяжении	DecisionTreeRegressor_pr	0.015367	0.895
Модуль упругости при растяжении	RandomForestRegressor_upr	0.00097	0.787
Модуль упругости при растяжении	LinearRegression_upr	0.00099	0.787
Модуль упругости при растяжении	DecisionTreeRegressor_upr	0.00108	0.787

Продолжение таблицы 2.

Модуль упругости при растяжении	KNeighborsRegressor_upr	0.001122	0.729
------------------------------------	-------------------------	----------	-------

2.3. Написать нейронную сеть, которая будет рекомендовать соотношение «матрица – наполнитель».

Для построения полносвязной нейросети переопределим X и y в соответствии с целевой переменной «Соотношение матрица-наполнитель». Разделим выборку на тренировочные и тестовые данные в соотношении 70/30 методом `train_test_split`.

В архитектуре ИНС используется модель `Sequential`. Она представляет собой линейный стек слоев. Модель состоит из 6 слоев.

На вход используем полносвязный слой `Dense`, количество нейронов = 16, активационная функция «`relu`».

Далее следуют четыре скрытых слоя – `BatchNormalization`, полносвязный `Dense`, слой `Dropout` (как метод регуляризации ИНС, предназначен для уменьшения переобучения сети за счет предотвращения сложных коадаптаций отдельных нейронов на тренировочных данных во время обучения) и еще один полносвязный слой `Dense`.

На выходе полносвязный слой `Dense` с одним нейроном, активационная функция «`sigmoid`».

```
#архитектура модели

model_ns = tf.keras.Sequential()

model_ns.add(Dense(16, input_dim=X_train.shape[1], activation = 'relu'))
model_ns.add(BatchNormalization())
model_ns.add(Dense(8, activation = 'relu'))
model_ns.add(Dropout(0.18))
model_ns.add(Dense(8, activation = 'relu'))
model_ns.add(Dense(1, activation = 'sigmoid'))
```

Рисунок 152 – Архитектура модели

Определим параметры, поищем оптимальные параметры, посмотрим на результаты. С помощью KerasClassifier выйдем на наилучшие параметры для нашей нейронной сети и построим окончательную нейросеть.

Model: "sequential"

Layer (type)	Output Shape	Param #
dense (Dense)	(None, 16)	208
batch_normalization (Batch Normalization)	(None, 16)	64
dense_1 (Dense)	(None, 8)	136
dropout (Dropout)	(None, 8)	0
dense_2 (Dense)	(None, 8)	72
dense_3 (Dense)	(None, 1)	9

=====
Total params: 489 (1.91 KB)
Trainable params: 457 (1.79 KB)
Non-trainable params: 32 (128.00 Byte)

Рисунок 16 – Построение нейросети

Обучим и оценим модель, посмотрим на потери, зададим функцию для визуализации факт/прогноз для результатов моделей.

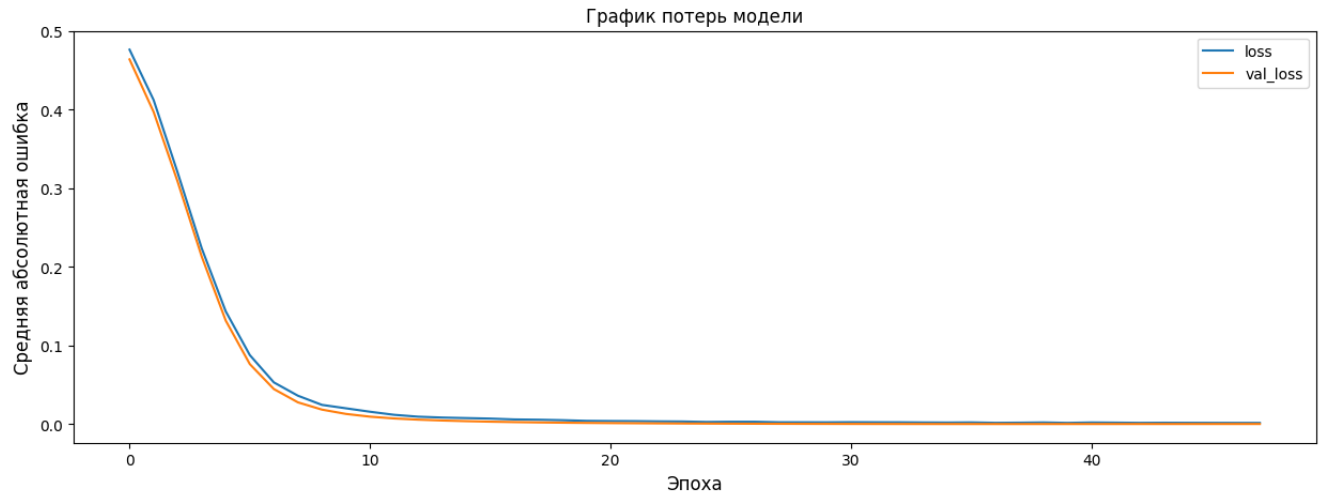


Рисунок 17 – График потерь

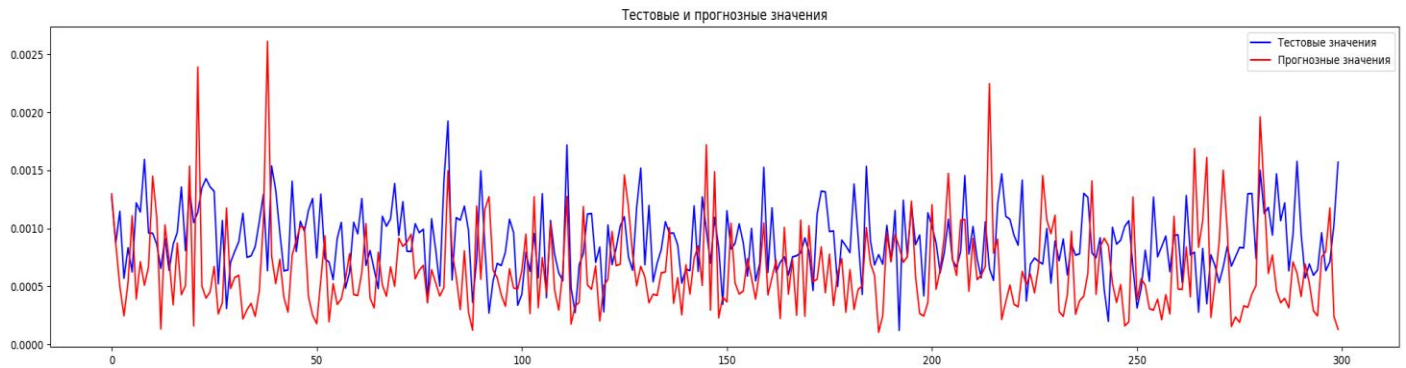


Рисунок 18 – Визуализация работы модели



Рисунок 19 – График прогнозных и настоящих значений

Разработка приложения

Приложение успешно работает и показывает результат прогноза для соотношения «Прочность при растяжении, МПа».

Для начала заходим в @BotFather в Telegram, создаем бота, и получаем его токен нашего бота, для связи телеграмма и исходного кода написанного на Py-Charm. Также прописываем название нашего бота, внутренние команды.

```
from aiogram import Bot, Dispatcher, executor, types

bot = Bot(token="YOUR_TOKEN")
dp = Dispatcher(bot)
```

Рисунок 20 – Пример соединения бота с TelegramBotAPI

Данное приложение — это основной файл DiplomBot.py где запускается чат-бот а также хендлеры бота, для обработок команд введенного пользователем.

```
1 usage
2
3 async def main(): # -> None
4     # Initialize Bot instance with a default parse mode which will be passed to all API calls
5     # bot = Bot(TOKEN, parse_mode=ParseMode.HTML)
6     # And the run events dispatching
7     dp.include_router(router)
8     await dp.start_polling(bot)
9
10
11 if __name__ == "__main__":
12     logging.basicConfig(level=logging.INFO, stream=sys.stdout)
13     try:
14         asyncio.run(main())
15     except KeyboardInterrupt:
16         print('Exit')
```

Рисунок 21 – Часть кода Diplombot.py

```
@router.message(CommandStart())
async def command_start_handler(message: Message):
    await message.answer(f"Привет, {hbold(message.from_user.full_name)}!")
    await message.answer('Я бот, который может предсказать\n-Модуль упругости при растяжении, ГПа\n-Прочность при '
        'растяжении, МПа\n-Соотношение матрица-наполнитель')

@router.message(Command('help'))
async def command_help(message: Message):
    await message.answer("Команда help")

@router.message(Command('req'))
async def req_one(message: Message, state: FSMContext):
    await state.set_state(Req.matrix)
    await message.answer('Введите Соотношение матрица-наполнитель')
```

Рисунок 21 – Часть кода handler.py

При запуске приложения, пользователь переходит на https://t.me/Raccococoon_eye_bot.

В открывшемся окне пользователю необходимо сначала команду /start, затем команду /req, далее бот предлагает какие признаки которые нужно ввести». Далее бот самостоятельно выводит результат и данные, которые пользователь ввел.

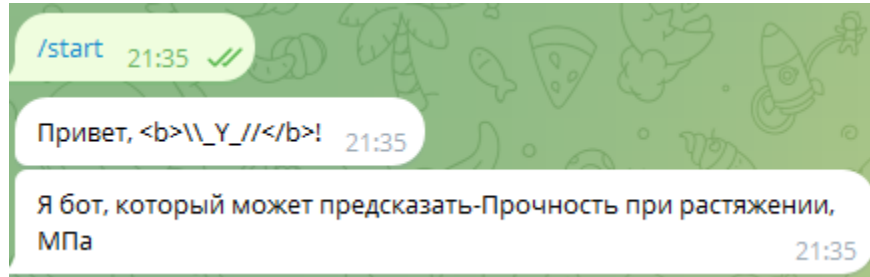


Рисунок 22 – Скриншот после ввода команды /start

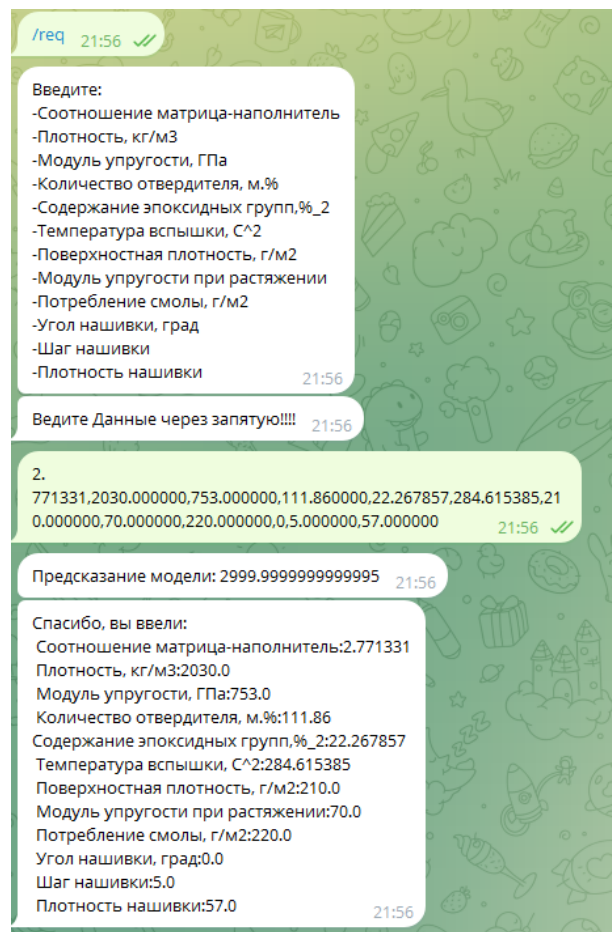


Рисунок 22 – Скриншот после ввода команды /req и ввода данных

На выходе пользователь получает результат прогноза для значения параметра «Прочность при растяжении, МПа».

2.4. Создание удалённого репозитория и загрузка

Для данного исследования был создан удаленный репозиторий на GitHub, который находится по адресу https://github.com/efremovyaroslav2015/DIPLOM_PROJECT. На него были загружены результаты работы: исследовательский notebook, код приложения.

Ноутбук с решением и приложением так же можно найти по адресу: <https://colab.research.google.com/drive/1UUbWMlnMvVV-gVflg5Y-mRb1jsBW85Ex?hl=ru#scrollTo=XlDXckXrSVbb>

Заключение

Этот поток операций и задач включает:

- Изучение теоретических методов анализа данных и машинного обучения;
- Изучение основ предметной области, в которой решается задача;
- Извлечение и трансформацию данных. Здесь нам был предоставлен готовый набор данных, поэтому через трудности работы с разными источниками и парсингом данных мы еще не соприкоснулись;
- Проведение разведочного анализа данных статистическими методами;
- DataMining — извлечение признаков из датасета и их анализ;
- Разделение имеющихся, в нашем случае размеченных, данных на обучающую, валидационную, тестовую выборки;
- Выполнение предобработки (препроцессинга) данных для обеспечения корректной работы моделей;
- Построение аналитического решения. Это включает выбор алгоритма решения и модели, сравнение различных моделей, подбор гиперпараметров модели;
- Визуализация модели и оценка качества аналитического решения;
- Сохранение моделей;
- Разработка и тестирование приложения для поддержки принятия решений специалистом предметной области, которое использовало бы найденную модель;
- Внедрение решения и приложения в эксплуатацию. Этот блок задач мы тоже пока не затронули.

Данная исследовательская работа позволяет сделать некоторые основные выводы по теме. Распределение полученных данных в объединённом датасете близко

к нормальному, но коэффициенты корреляции между парами признаков стремятся к нулю.

Результат при нормализации данных дал отличные результаты при обучении моделей.

Использованные при разработке моделей подходы не позволили получить сколько-нибудь достоверных прогнозов. Применённые модели регрессии показали высокой эффективности в прогнозировании свойств композитов. Лучшие метрики для модуля упругости при растяжении, ГПа – метод опорных векторов, для прочности при растяжении, МПа – Случайный лес. Была проведена обучение нейронной сети из свойств материалов соотношение «матрица – наполнитель». Что тоже показала отличные результаты.

Из-за первоначальной отсутствие корреляции между признаками, с помощью анализа данных и грамотного предпроцессинга дали отличные результаты.

2.5. Список используемой литературы и веб ресурсы.

1. 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: 07.06.2022)
2. Andre Ye. 5 алгоритмов регрессии в машинном обучении, о которых вам следует знать: – Режим доступа: <https://habr.com/ru/company/vk/blog/513842/> (дата обращения: 01.06.2022).
3. Devpractice Team. Python. Визуализация данных. Matplotlib. Seaborn. Mayavi. - devpractice.ru. 2020. - 412 с.: ил.
4. Абросимов Н.А.: Методика построения разрешающей системы уравнений динамического деформирования композитных элементов конструкций (Учебно-методическое пособие), ННГУ, 2010
5. Абу-Хасан Махмуд, Масленникова Л. Л.: Прогнозирование свойств композиционных материалов с учётом наноразмера частиц и акцепторных свойств катионов твёрдых фаз, статья 2006 год
6. Бизли Д. Python. Подробный справочник: учебное пособие. – Пер. с англ. – СПб.: Символ-Плюс, 2010. – 864 с., ил.
7. 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.
8. Грас Д. Data Science. Наука о данных с нуля: Пер. с англ. - 2-е изд., перераб. и доп. - СПб.: БХВ-Петербург, 2021. - 416 с.: ил.
9. Документация по библиотеке keras: – Режим доступа: <https://keras.io/api/>. (дата обращения: 08.06.2022).
10. Документация по библиотеке matplotlib: – Режим доступа: <https://matplotlib.org/stable/users/index.html>. (дата обращения: 10.06.2022)
11. Документация по библиотеке numpy: – Режим

доступа: <https://numpy.org/doc/1.22/user/index.html#user>. (дата обращения: 03.06.2022).

12. Документация по библиотеке pandas: – Режим доступа: https://pandas.pydata.org/docs/user_guide/index.html#user-guide. (дата обращения: 04.06.2022).

13. Документация по библиотеке scikit-learn: – Режим доступа: https://scikit-learn.org/stable/user_guide.html. (дата обращения: 05.06.2022).

14. Документация по библиотеке seaborn: – Режим доступа: <https://seaborn.pydata.org/tutorial.html>. (дата обращения: 06.06.2022).

15. Документация по библиотеке Tensorflow: – Режим доступа: <https://www.tensorflow.org/overview> (дата обращения: 10.06.2022).

16. Документация по языку программирования python: – Режим доступа: <https://docs.python.org/3.8/index.html>. (дата обращения: 02.06.2022).

17. Иванов Д.А., Ситников А.И., Шляпин С.Д – Композиционные материалы: учебное пособие для вузов, 2019. 13 с.

18. Краткий обзор алгоритма машинного обучения Метод Опорных Векторов (SVM) – Режим доступа: <https://habr.com/ru/post/428503/> (дата обращения 07.06.2022)

19. Ларин А. А., Способы оценки работоспособности изделий из композиционных материалов методом компьютерной томографии, Москва, 2013, 148 с.

20. Материалы конференции: V Всероссийская научно-техническая конференция «Полимерные композиционные материалы и производственные технологии нового поколения», 19 ноября 2021 г.

21. Миронов А.А. Машинное обучение часть I ст.9 – Режим доступа: <http://is.ifmo.ru/verification/machine-learning-mironov.pdf>. (дата обращения 08.06.2022)

22. Плас Дж. Вандер, Python для сложных задач: наука о данных и

машинное обучение. Санкт-Петербург: Питер, 2018, 576 с.

23. Реутов Ю.А.: Прогнозирование свойств полимерных композиционных материалов и оценка надёжности изделий из них, Диссертация на соискание учёной степени кандидата физико-математических наук, Томск 2016.

24. Роббинс, Дженнифер. HTML5: карманный справочник, 5-е издание.: Пер. с англ. - М.: ООО «И.Д. Вильямс»: 2015. - 192 с.: ил.

25. Руководство по быстрому старту в flask: – Режим доступа: <https://flask-russian-docs.readthedocs.io/ru/latest/quickstart.html>. (дата обращения: 09.06.2022)

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

27. Скиена, Стивен С. С42 Наука о данных: учебный курс.: Пер. с англ. - СПб.: ООО "Диалектика", 2020. - 544 с. : ил.

28. Справочник по композиционным материалам: в 2 - х кн. Кн. 2 / Под ред. Дж. Любина; Пер. с англ. Ф. Б. Геллера, М. М. Гельмонта; Под ред. Б. Э. Геллера - М.: Машиностроение, 1988. - 488 с. : ил;

29. Траск Эндрю. Грокаем глубокое обучение. – СПб.: Питер, 2019. – 352 с.: ил.

30. Чун-Те Чен и Грейс Х. Гу. Машинное обучение для композитных материалов (март 2019г.) – Режим доступа: <https://www.cambridge.org/core/journals/mrs-communications/article/machine-learning-for-composite-materials/F54F60AC0048291BA47E0B671733ED15>. (дата обращения 02.06.2022)