

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет имени Н.Э. Баумана  
(национальный исследовательский университет)»

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**  
**по курсу**  
**«Data Science»**

Слушатель

Савичев Павел Михайлович

Москва, 2022

## Содержание

Содержание.....	2
Введение.....	3
1. Аналитическая часть.....	5
1.1. Постановка задачи .....	5
1.2. Разведочный анализ данных .....	7
2. Практическая часть.....	20
2.1. Предобработка данных.....	20
2.2. Разработка и тестирование моделей .....	24
2.3. Нейронная сеть, рекомендации соотношения матрица- наполнитель .....	30
Заключение .....	31
Список используемой литературы и ссылки на веб-ресурсы .....	32

## Введение

Тема Выпускной квалификационной работы: Прогнозирование конечных свойств новых материалов (композиционных материалов).

Актуальность темы исследования: Изделия из полимерных композиционных материалов применяются во многих секторах российской и мировой экономики: машиностроении, строительстве, радиоэлектронике и др. Работоспособность и качество изделий зависит от многих факторов, в том числе от физико-механических свойств материалов, из которых они изготовлены.

Ежегодно разрабатываются десятки и сотни новых полимерных композиционных материалов. На практике для изучения эксплуатационных свойств полимерных композиционных материалов необходимо задействовать методы экспериментальных исследований, для этого необходимо найти высококвалифицированных сотрудников, закупить и обслуживать высокоточное научно-исследовательское и лабораторное оборудование. При этом следует учитывать, что экспериментальные исследования в условиях реального производства требуют значительных временных затрат, материальных и человеческих ресурсов.

Одним из видов модификации полимеров или направленного изменения физико-механических свойств является структурное модифицирование, т.е. изменение физико-механических свойств полимера без изменения его химического состава и его молекулярной массы.

Структурное модифицирование полимеров обычно осуществляют в процессе переработки полимеров регулированием параметров формирования изделия, например температуры и времени процесса, режимов нагревания и охлаждения при переработке из расплава или природы растворителя и условий его удаления при переработке из раствора, а также введением в полимер небольшого количества веществ, воздействующих на кинетику образования полимерного тела и (или) морфологию полимера.

В основе структурного модифицирования полимеров лежит многообразие сосуществующих в полимере структурных форм и взаимосвязь их морфологии с условиями формирования полимерного тела.

Оценка надежности конструкций по критериям прочности основана на сопоставлении параметров напряженно-деформированного состояния и соответствующих предельных прочностных свойств материалов элементов конструкций.

Создание новых ПКМ и оценка надежности конструкций из них требует нового подхода в определении статистических деформационно-прочностных и теплофизических характеристик как основы оценки надежности конструкций, а также изучения возможности применения средств и методов вычислительной механики для моделирования эффективных статистических характеристик наполненных композиций и расчета вероятности безотказной работы изделий.

Как следствие, в настоящее время начали применять технологии машинного обучения для изучения свойств композиционных материалов, так же проводятся исследования влияния одной или нескольких независимых переменных на зависимую переменную.

Актуальность решения этой задачи тесно связано с активным использованием строительных и инновационных композитных материалов (технологий) во многих областях производства.

Полученный прогноз модели должен помочь значительно сократить количество экспериментальных исследований, а также пополнить базу данных материалов возможными новыми характеристиками материалов, и цифровыми двойниками новых композитов.

# 1. Аналитическая часть

## 1.1 Постановка задачи.

**Описание:** Композиционные материалы или композиты – это материалы, состоящие из двух и более компонентов (отдельных волокон или других армирующих составляющих и связующей их матрицы) и обладающие специфическими свойствами, отличными от суммарных свойств составляющих их компонентов. Компоненты композитов не должны растворяться или иным способом поглощать друг друга. Они должны быть хорошо совместимы. Свойства композиционных материалов нельзя определить только по свойствам компонентов, без учета их взаимодействия.

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

**На входе** имеются данные о начальных свойствах компонентов композиционных материалов (количество связующего, наполнителя, температурный режим отверждения и т.д.).

Первый датасет «X\_br.xlsx» включает в себя десять переменных Базальтопластика, такие как соотношение матрица-наполнитель, плотность, модуль упругости, количество отвердителя, содержание эпоксидных групп, температура вспышки, поверхностная плотность, модуль упругости при растяжении, прочность при растяжении, потребление смолы. Выборка содержит 1023 замера.

Второй датасет «X\_npr.xlsx» включает в себя три переменных накладок углепластиковых – угол нашивки, шаг нашивки, плотность нашивки. Выборка содержит 1040 замеров.

Данные таблицы имеют колонку с целочисленным индексом, не являющимся входным или выходным переменным, служащим для сопоставления таблиц данных.

**На выходе** необходимо спрогнозировать три параметра: модуля упругости при растяжении, прочности при растяжении, соотношение матрица-наполнитель.

Поставленная задача в рамках классификации категорий машинного обучения относится к машинному обучению с учителем, задача регрессии.

Анализ, предобработка данных, построение моделей выполнены посредством языка программирования Python с использованием библиотек Pandas, Matplotlib и Sklearn.

## 1.2 Разведочный анализ данных.

Целями разведочного анализа является 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.

На первом этапе делаем объединение по индексу (так как других общих признаков нет) двух датасетов в один. Объединение выполняем по типу INNER. Видим, что размер датасетов не одинаковый. Разница составляет 17 строк. Принимаем решение разницу в 17 строк отбросить, чтобы уравнивать размер датасетов.

Далее делаем проверку объединённого датасета на наличие пропусков и определяем тип данных.

```
df_merge.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, C_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 1 – Информация о пропусках и типе данных

Полученный датасет имеет 1023 записи (10 входных параметров и 3 выходных параметра). Пропуски в полученном датасете отсутствуют, тип данных float64 (число с плавающей точкой) и int64 (целые числа) - информация о типе данных представлена на Рисунке 1.

Далее необходимо выполнить проверку на наличие повторений (дубликатов данных). Дубликаты в объединённом датафрейме не были найдены – это видно на рисунке 2.

```
df_merge.duplicated().sum()
```

0

Рисунок 2 – Наличие дубликатов записей

На следующем этапе сделаем визуализацию данных. Чтобы получить наглядное представление о характерах распределений переменных необходимо построить гистограммы и диаграммы размаха «ящик с усами» (такой вид диаграммы в удобной форме показывает медиану, нижний и верхний квартили, минимальное и максимальное значение выборки и выбросы).

Производим отрисовку гистограммы распределения для каждой переменной.

```
for col in df_merge.columns:  
    plt.figure(figsize=(15, 5))  
    sns.histplot(data=df_merge[col])  
    plt.show()
```

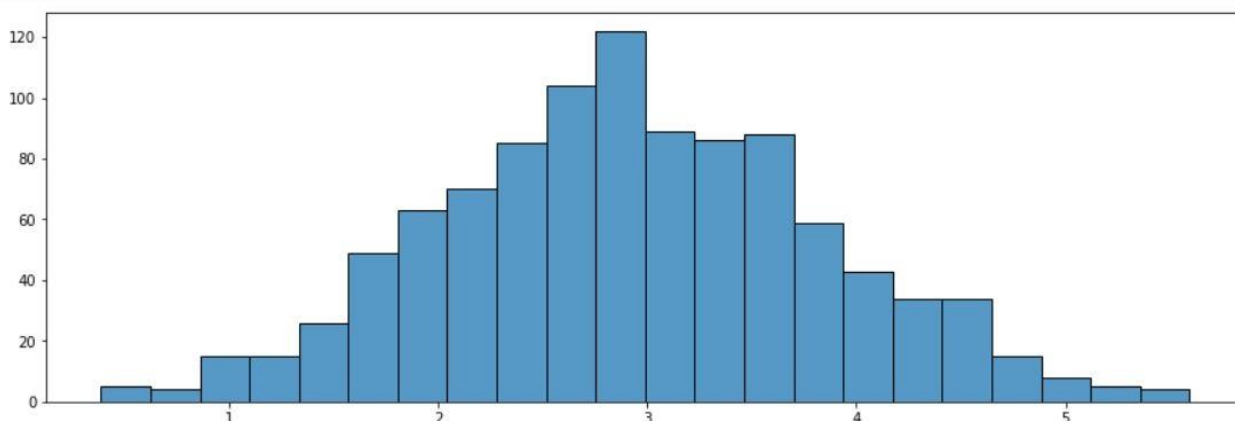


Рисунок 3 – Гистограмма соотношения матрицы-наполнителя



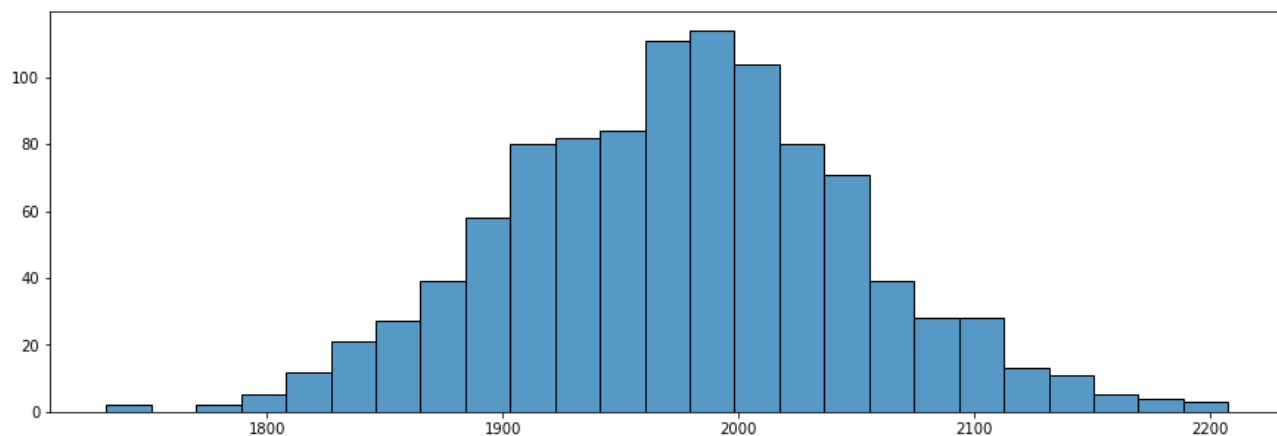


Рисунок 4 – Гистограмма плотности, кг/м<sup>3</sup>

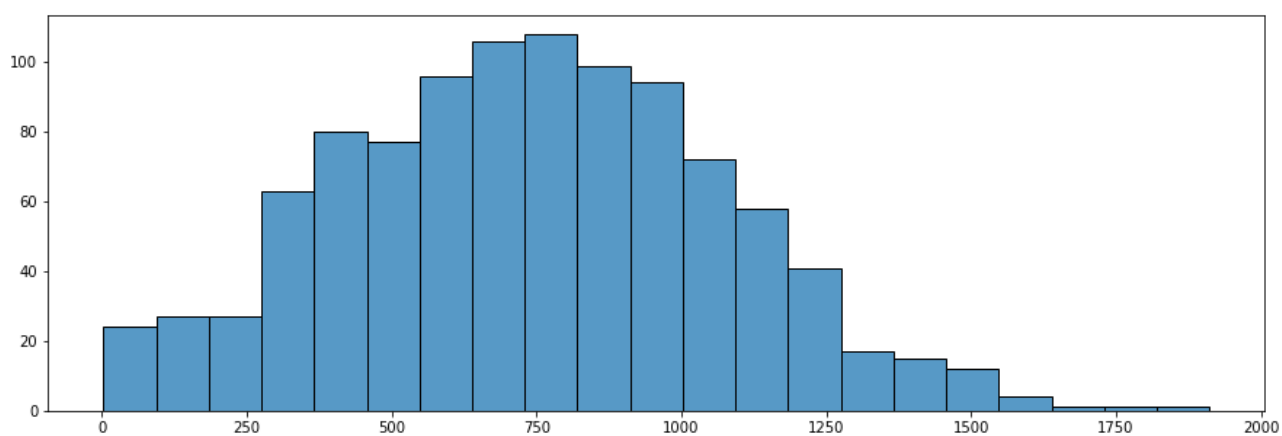


Рисунок 5 – Гистограмма модуля упругости, ГПа

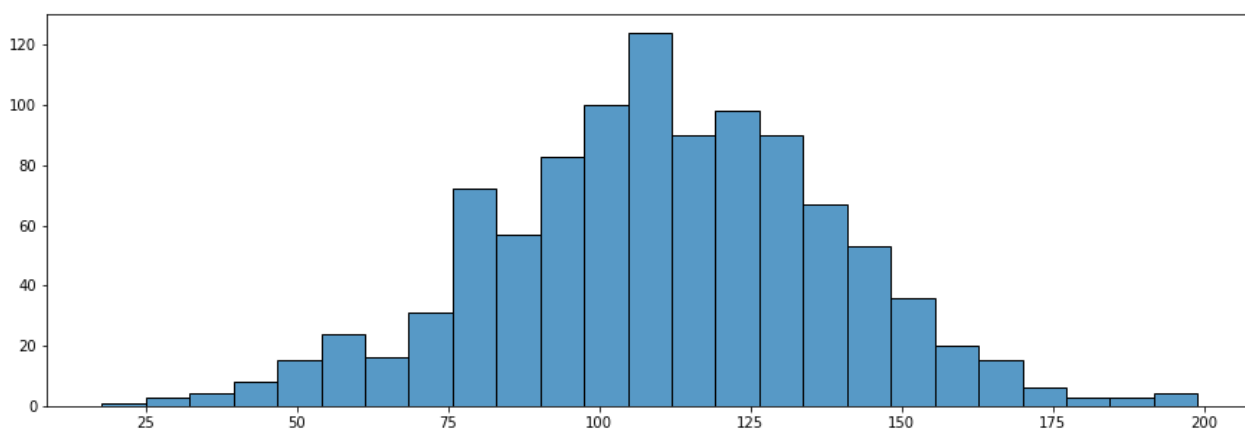


Рисунок 6 – Гистограмма количества отвердителя, м.%

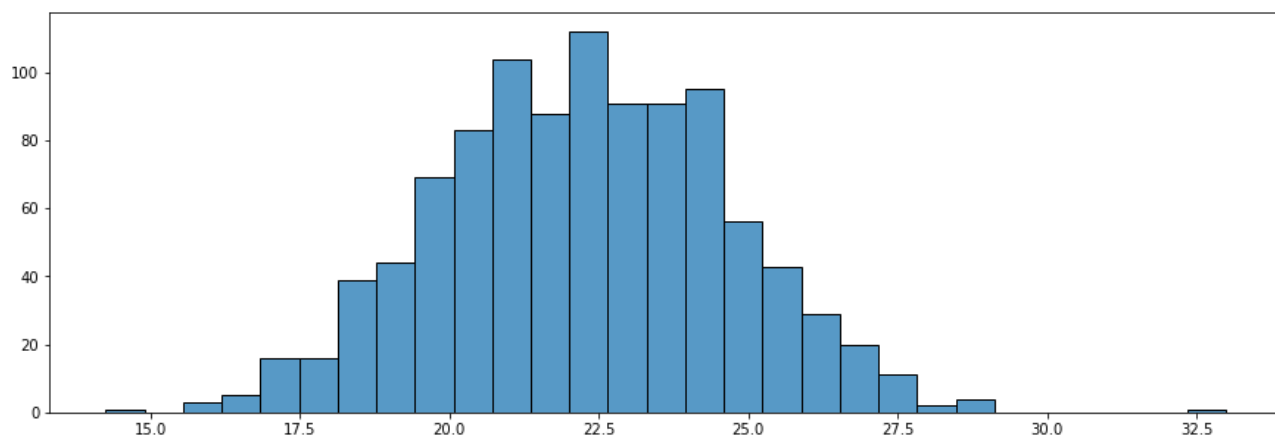


Рисунок 7 – Гистограмма содержания эпоксидных групп, %\_2

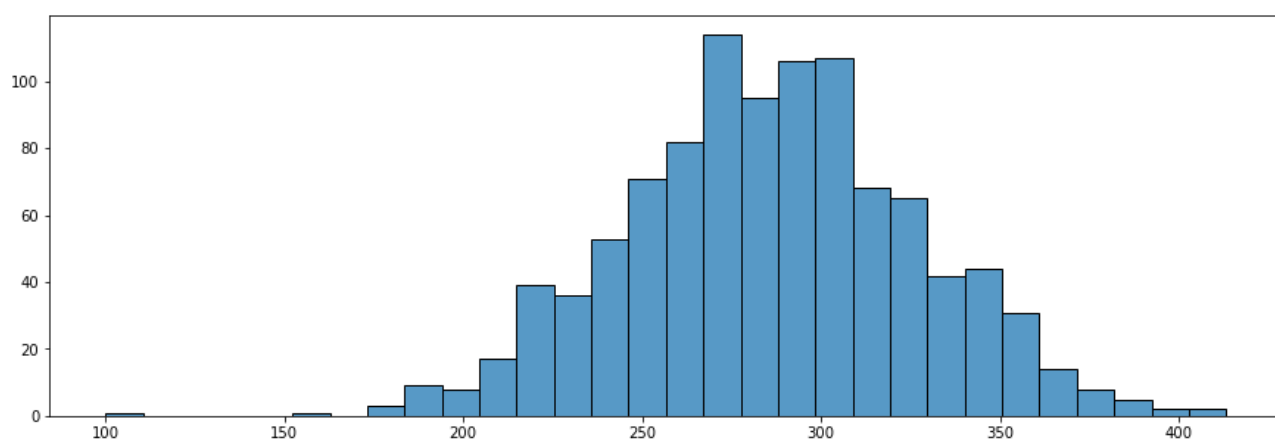


Рисунок 8 – Гистограмма температуры вспышки, C\_2

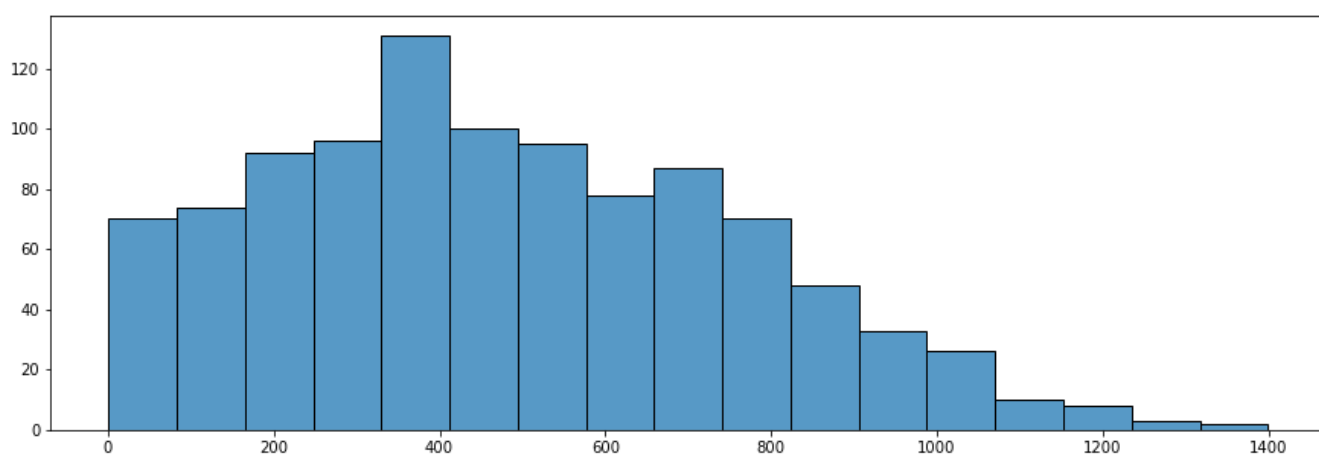


Рисунок 9 – Гистограмма поверхностной плотности, г/м2

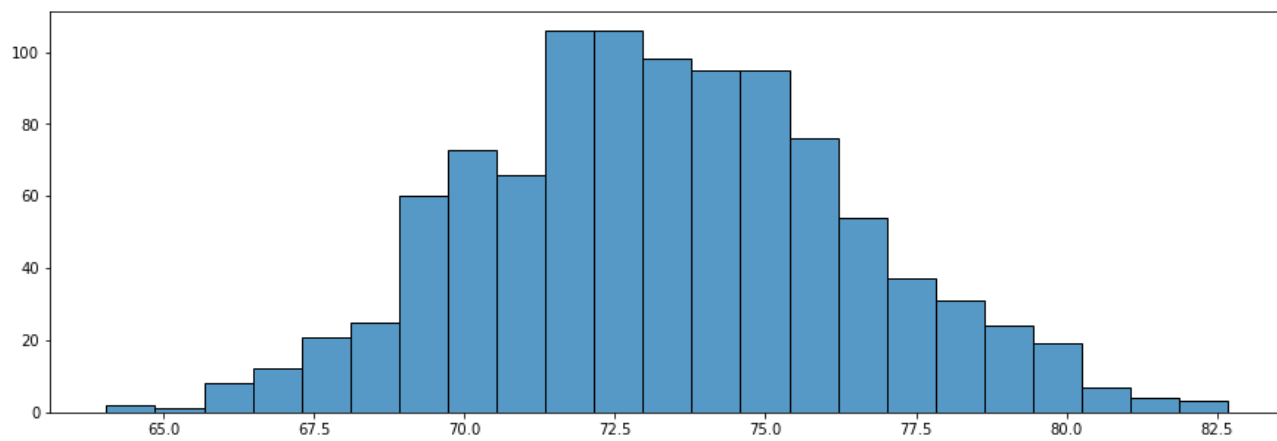


Рисунок 10 – Гистограмма модуля упругости при растяжении, ГПа

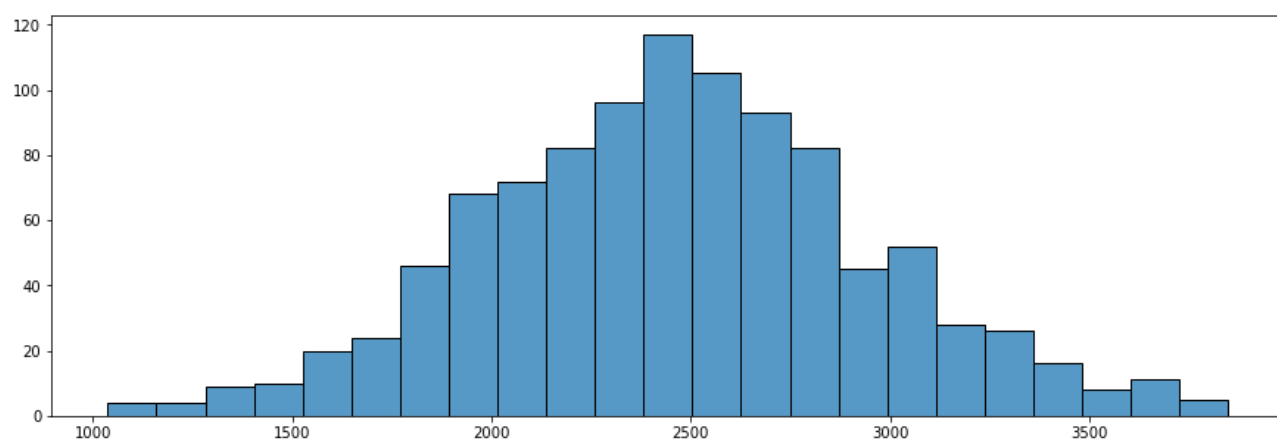


Рисунок 11 – Гистограмма прочности при растяжении, МПа

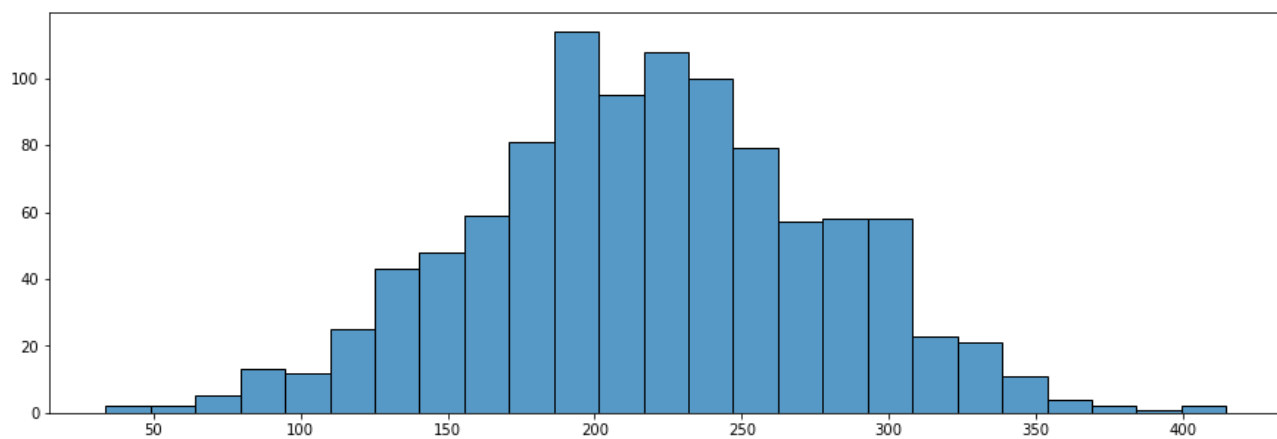


Рисунок 12 – Гистограмма потребления смолы, г/м2

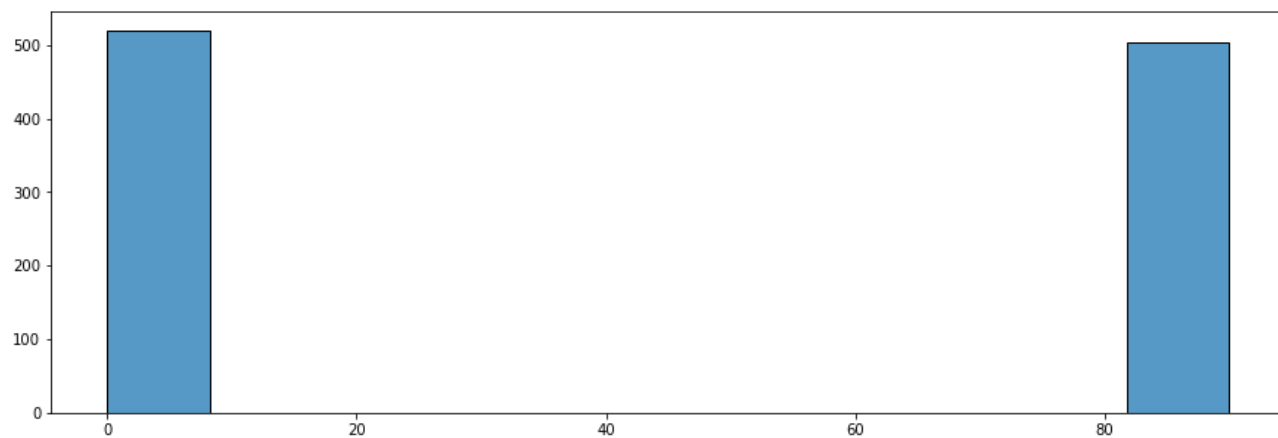


Рисунок 13 – Гистограмма угла нашивки, град

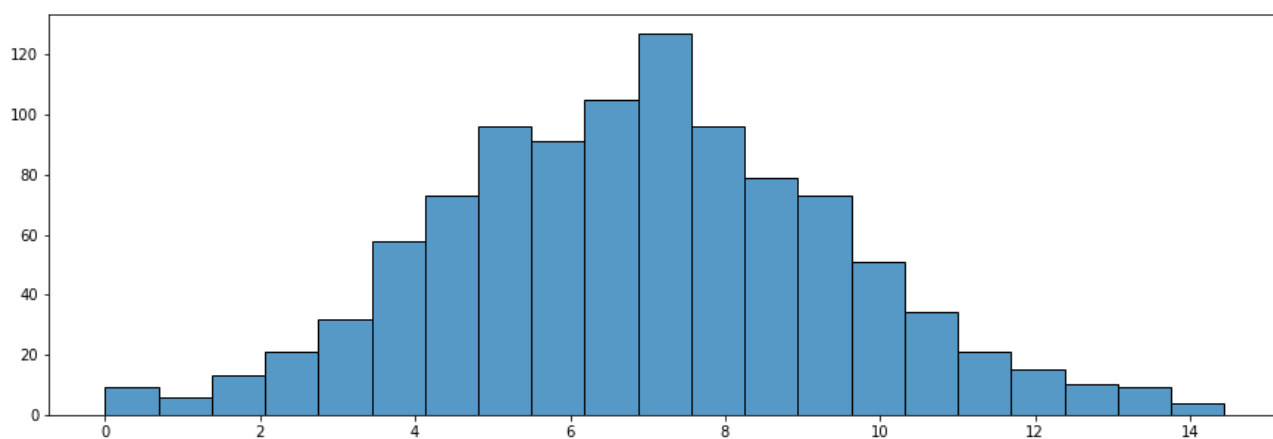


Рисунок 14 – Гистограмма шага нашивки

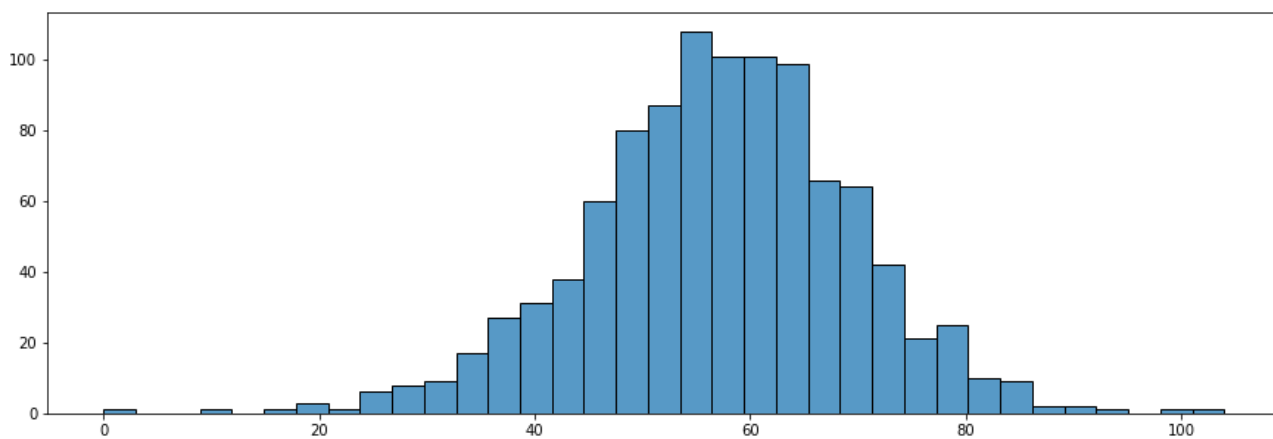


Рисунок 15 – Гистограмма плотности нашивки

В соответствии с рисунками 3-12,14,15 распределение нормальное, имеются выбросы. Яркие выраженные выбросы содержатся в гистограммах плотность в соответствии с рисунком 4, количества отвердителя в соответствии с рисунком 6, содержание эпоксидных групп в соответствии с рисунком 7, температура вспышки в соответствии с рисунком 8, плотность нашивки растяжении в соответствии с рисунком 15. Гистограмма угла нашивки имеет бинарное значение в соответствии с рисунком 13.

Вторым шагом построим диаграммы «ящик с усами» для каждого признака. Это поможет нам определить все выбросы и избавиться от них в дальнейшем для того, чтобы набор данных имел более сглаженный вид с точки зрения нормализации.

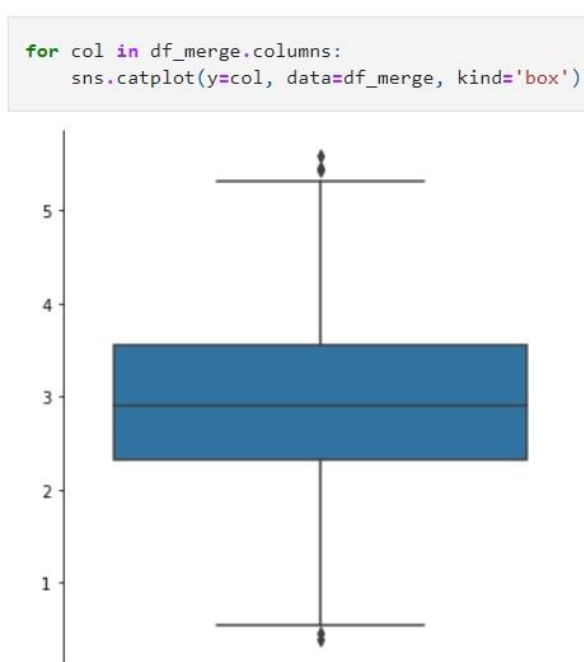


Рисунок 16 – График соотношения матрицы-наполнителя

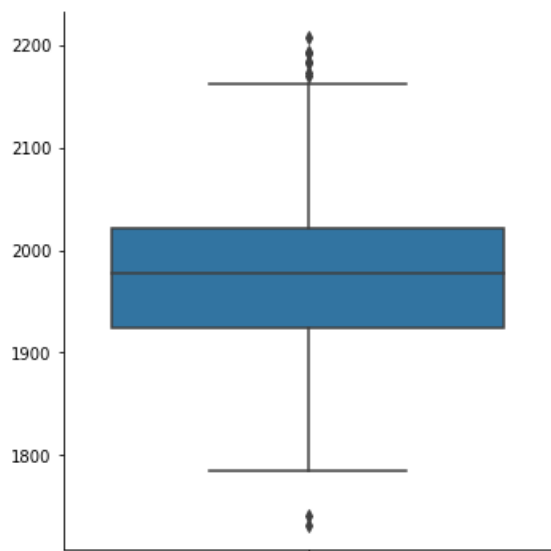


Рисунок 17 – График плотности,  
кг/м3

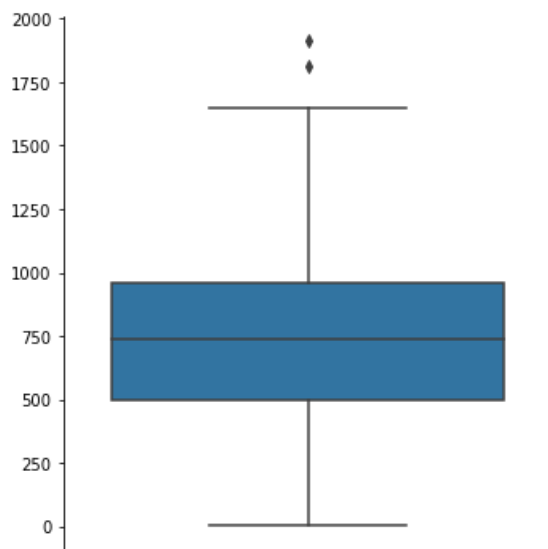


Рисунок 18 – График модуля упруго-  
сти, ГПа

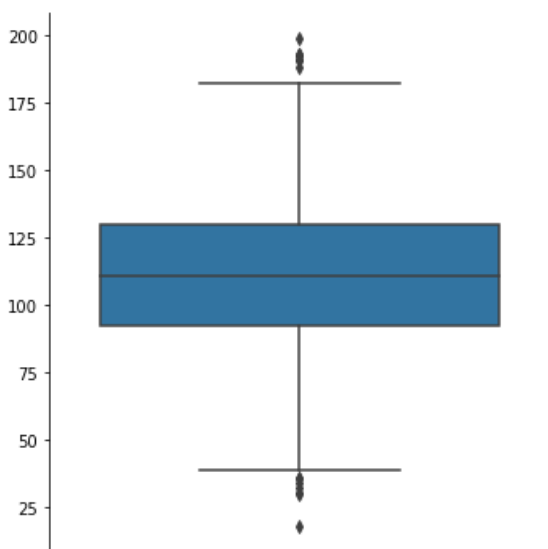


Рисунок 19 – График количества  
отвердителя, м.%

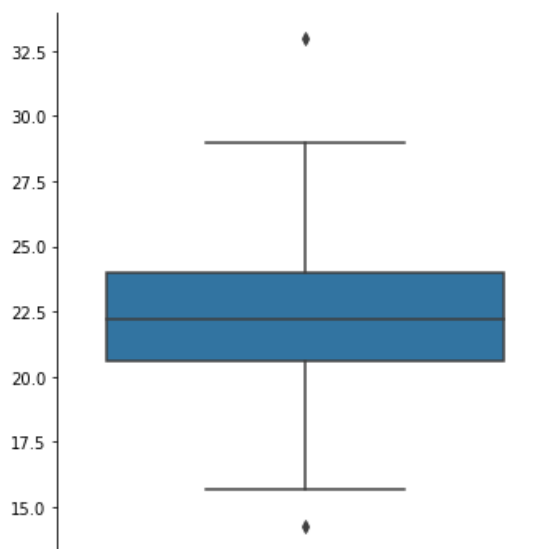


Рисунок 20 – График содержания  
эпоксидных групп, %\_2

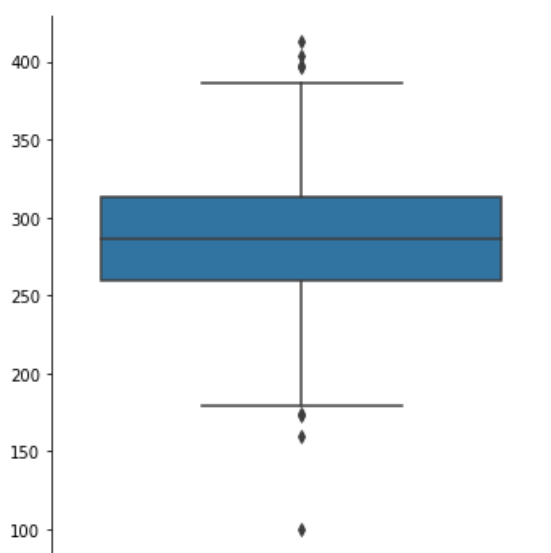


Рисунок 21 – График температуры вспышки,  $C_2$

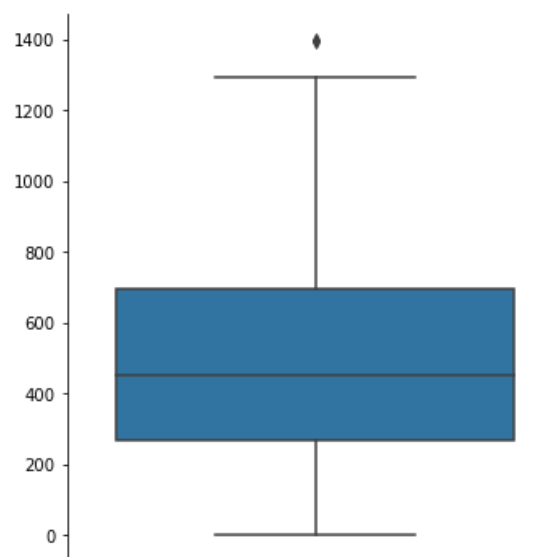


Рисунок 22 – График поверхностной плотности,  $г/м^2$

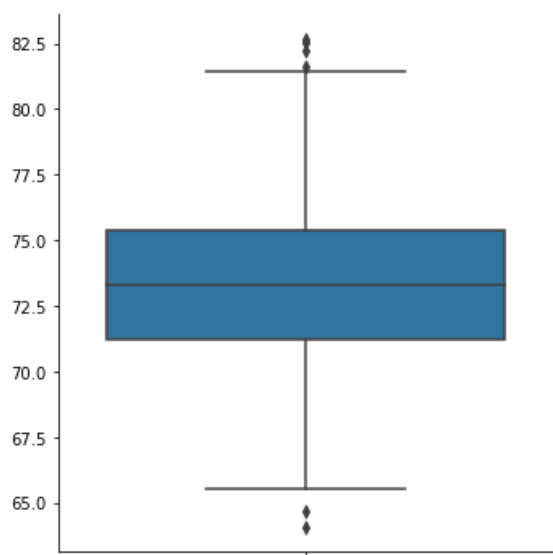


Рисунок 23 – График модуля упругости при растяжении,  $ГПа$

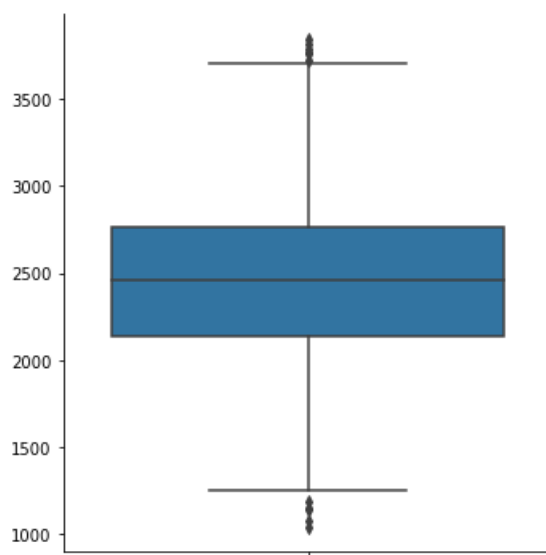


Рисунок 24 – График прочности при растяжении,  $МПа$

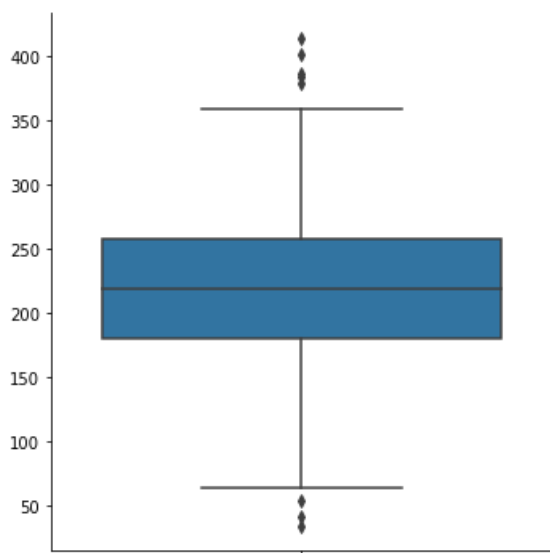


Рисунок 25 – График потребления  
смолы, г/м2

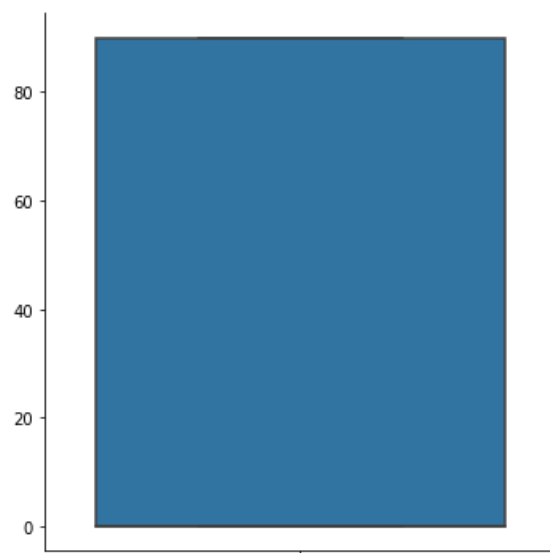


Рисунок 26 – График угла нашивки,  
град

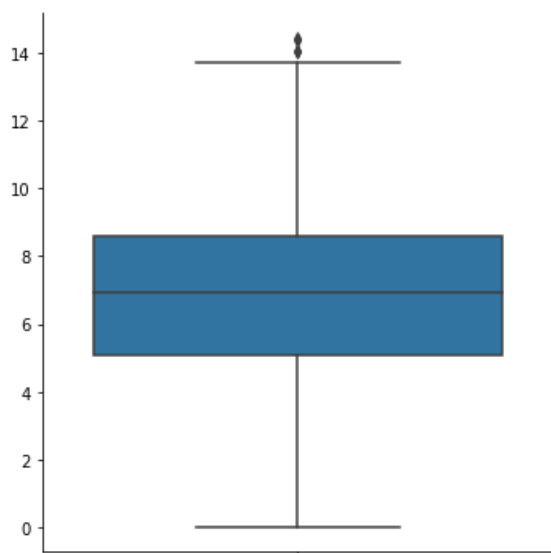


Рисунок 27 – График шага нашивки

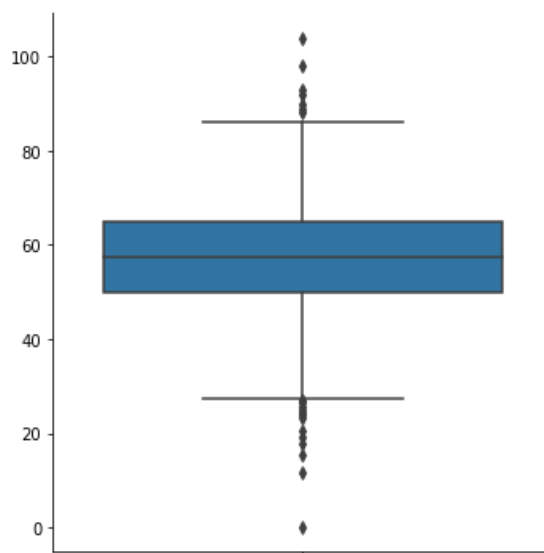


Рисунок 28 – График плотность  
нашивки



Диаграммы «Ящик с усами» показали, что у всех признаков в соответствии с рисунками 16-25, 27,28 имеются выбросы. Выбросы не имеют экстремально больших отклонений. График «Угол нашивки» не имеет выбросов, так как имеет 2 значения (0 градусов и 90 градусов), в соответствии с рисунком 26.

Следующим шагом разведочного анализа построим попарные графики рассеяния точек в соответствии с рисунком 29 и тепловую карту матрицы корреляции в соответствии с рисунком 30, для визуализации наличия зависимости признаков.

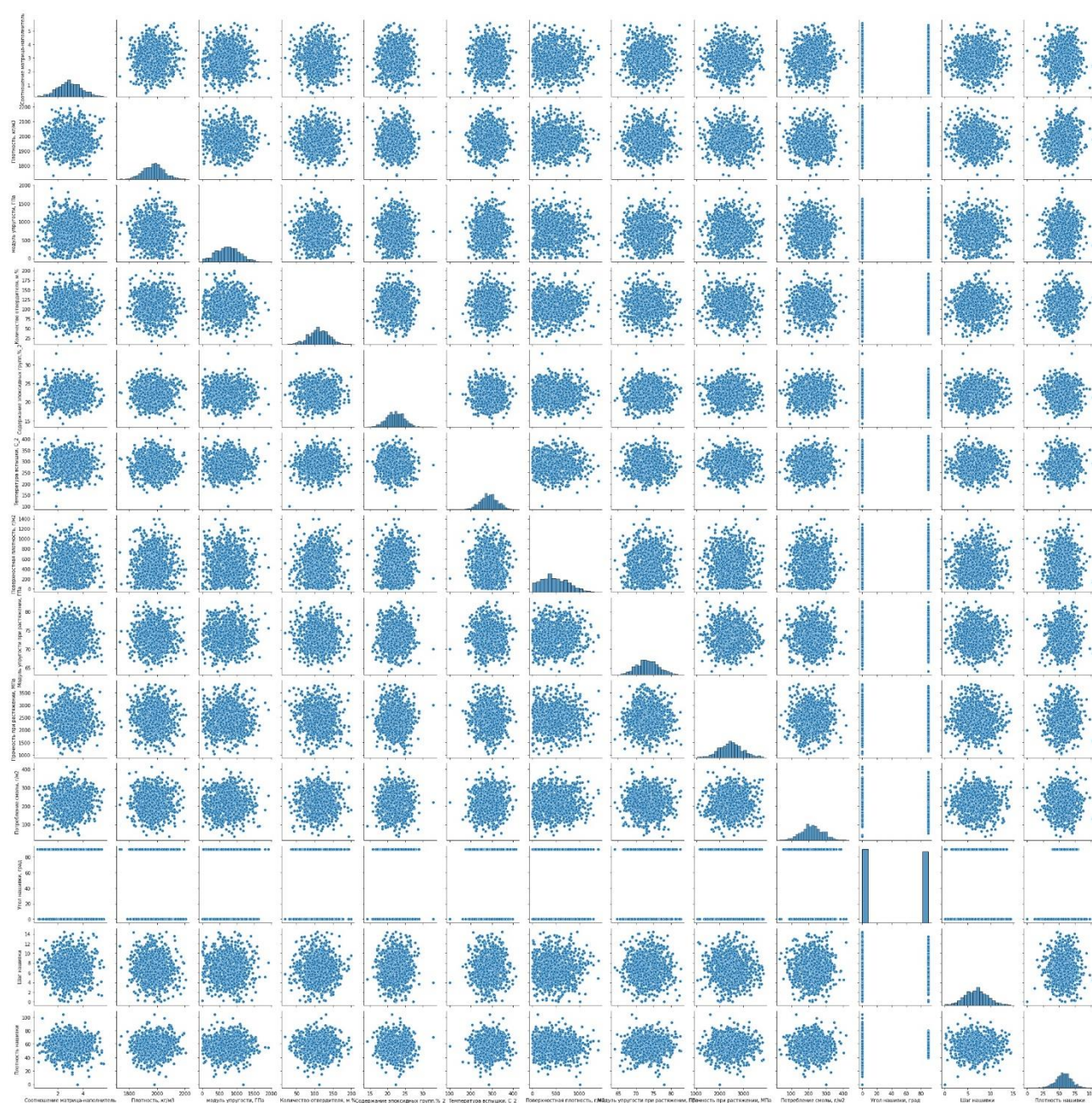


Рисунок 29 – График попарного рассеяния точек

```
corrmat = df_merge.corr()
f, ax = plt.subplots(figsize =(15, 12))
sns.heatmap(corrmat, ax = ax, cmap ="YlGnBu", linewidths = 0.1, annot = True)
```

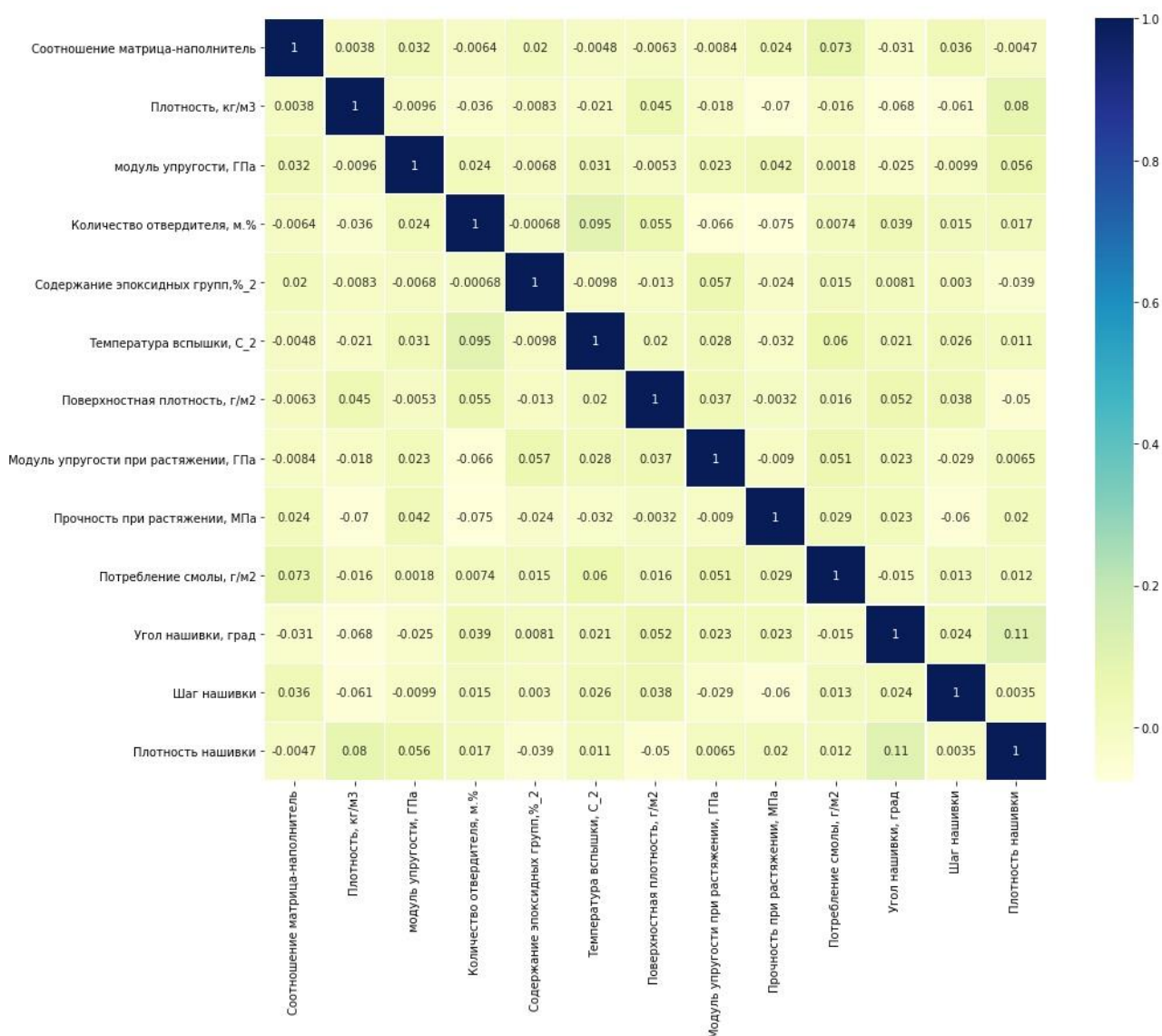


Рисунок 30 – Тепловая карта матрицы корреляции

Как видно по рисунку 30, попарное сравнение признаков, зависимости не выявило. Также корреляционная матрица, тоже не выявила каких-либо зависимостей. Зависимость между признаками очень низкая. Самая высокая зависимость между углом нашивки и плотность нашивки (0,11).

Финальным шагом разведочного анализа посчитаем среднее в соответствии с рисунком 31 и медианное значение в соответствии с рисунком 32, для каждого признака.

df_merge.mean()	
Соотношение матрица-наполнитель	2.930366
Плотность, кг/м3	1975.734888
модуль упругости, ГПа	739.923233
Количество отвердителя, м.%	110.570769
Содержание эпоксидных групп,%_2	22.244390
Температура вспышки, С_2	285.882151
Поверхностная плотность, г/м2	482.731833
Модуль упругости при растяжении, ГПа	73.328571
Прочность при растяжении, МПа	2466.922843
Потребление смолы, г/м2	218.423144
Угол нашивки, град	44.252199
Шаг нашивки	6.899222
Плотность нашивки	57.153929
dtype: float64	

Рисунок 31 – Среднее значение признаков

df_merge.median()	
Соотношение матрица-наполнитель	2.906878
Плотность, кг/м3	1977.621657
модуль упругости, ГПа	739.664328
Количество отвердителя, м.%	110.564840
Содержание эпоксидных групп,%_2	22.230744
Температура вспышки, С_2	285.896812
Поверхностная плотность, г/м2	451.864365
Модуль упругости при растяжении, ГПа	73.268805
Прочность при растяжении, МПа	2459.524526
Потребление смолы, г/м2	219.198882
Угол нашивки, град	0.000000
Шаг нашивки	6.916144
Плотность нашивки	57.341920
dtype: float64	

Рисунок 32 – Медианное значение признаков

В результат проведенного разведочного анализа мы можем сделать следующие выводы. У почти у всех признаков имеется нормальное распределение и по имеющейся информации, данные являются предварительно обработанными заказчиком. Пропуски в заполнении данных отсутствуют. Взаимозависимость признаков почти полностью отсутствует. Очень слабая зависимость есть между углом нашивки и плотность нашивки. Исходные датасеты имеют разный размер, но так как разница в размере выборки составляет всего 17 замеров или 1,66 процента от размера выборки «X\_br.xlsx», я отбросил лишние данные и произвел объединение по типу inner.



## 2. Практическая часть

### 2.1 Предобработка данных

Для начала, посчитаем количество выбросов. В соответствии с моим предположением выбросов должно быть не много, исходя из диаграмм «ящик усами». Для этого посчитаем количество выбросов двумя основными способами. Методом 3-х сигм и методом межквартильных расстояний.

```
count_3s = 0
count_iq = 0
for column in df_merge:
    d = df_merge.loc[:, [column]]
    # методом 3-х сигм
    zscore = (df_merge[column] - df_merge[column].mean()) / df_merge[column].std
    d['3s'] = zscore.abs() > 3
    count_3s += d['3s'].sum()
    # методом межквартильных расстояний
    q1 = np.quantile(df_merge[column], 0.25)
    q3 = np.quantile(df_merge[column], 0.75)
    iqr = q3 - q1
    lower = q1 - 1.5 * iqr
    upper = q3 + 1.5 * iqr
    d['iq'] = (df_merge[column] <= lower) | (df_merge[column] >= upper)
    count_iq += d['iq'].sum()
print('Метод 3-х сигм, выбросов:', count_3s)
print('Метод межквартильных расстояний, выбросов:', count_iq)
```

Метод 3-х сигм, выбросов: 24

Метод межквартильных расстояний, выбросов: 93

Рисунок 33 – Количество выбросов

В соответствии с рисунком 33, метод 3-х сигм нашел меньше выбросов 24 выброса, против 93 у метода межквартильных расстояний. Учитывая тот факт, что данные были предварительно подготовлены заказчиком и то, что график "ящик с усами" показывает небольшое количество выбросов и не самый большой размах. С целью того, чтобы избежать удаления тех данных, которые могут оказаться не выбросами, а особенностями датасета, я оставил свой выбор за методом 3-х сигм.

Для того чтобы не допустить ошибки и не удалить особенности признака. Посчитаем распределение выбросов по признакам в соответствии с рисунком 34.

```
t_df = df_merge.copy()
for i in df_merge.columns:
    t_df[i] = abs((df_merge[i] - df_merge[i].mean()) / df_merge[i].std())
    print(f"{sum(t_df[i] > 3)} выбросов в признаке {i}")
print(f'Всего {sum(sum(t_df.values > 3))} выброса')
```

```
0 выбросов в признаке Соотношение матрица-наполнитель
3 выбросов в признаке Плотность, кг/м3
2 выбросов в признаке модуль упругости, ГПа
2 выбросов в признаке Количество отвердителя, м.%
2 выбросов в признаке Содержание эпоксидных групп,%_2
3 выбросов в признаке Температура вспышки, С_2
2 выбросов в признаке Поверхностная плотность, г/м2
0 выбросов в признаке Модуль упругости при растяжении, ГПа
0 выбросов в признаке Прочность при растяжении, МПа
3 выбросов в признаке Потребление смолы, г/м2
0 выбросов в признаке Угол нашивки, град
0 выбросов в признаке Шаг нашивки
7 выбросов в признаке Плотность нашивки
Всего 24 выброса
```

Рисунок 34 – Распределение выбросов по признакам

Как видно из расчёта в соответствии с рисунком 34, выбросы распределены по разным признакам. Нет какой-либо чрезмерной концентрации в одном признаке. Соответственно можно приступить к удалению признака, так как существенных изменений на зависимости они не окажут.

Следующим шагом произведем удаление выбросов из датасета и проверку изменения количества строк.

```
clean_outliers_df_merge = df_merge[(np.abs(stats.zscore(df_merge)) <= 3).all(axis=1)]
clean_outliers_df_merge
```

	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
1	1.857143	2030.000000	738.736842	50.000000	23.750000	284.615385	210.000000	70.000000	3000.000000	220.000000	0
3	1.857143	2030.000000	738.736842	129.000000	21.250000	300.000000	210.000000	70.000000	3000.000000	220.000000	0
4	2.771331	2030.000000	753.000000	111.860000	22.267857	284.615385	210.000000	70.000000	3000.000000	220.000000	0
5	2.767918	2000.000000	748.000000	111.860000	22.267857	284.615385	210.000000	70.000000	3000.000000	220.000000	0
6	2.569620	1910.000000	807.000000	111.860000	22.267857	284.615385	210.000000	70.000000	3000.000000	220.000000	0
...	...	...	...	...	...	...	...	...	...	...	...
1018	2.271346	1952.087902	912.855545	86.992183	20.123249	324.774576	209.198700	73.090961	2387.292495	125.007669	90
1019	3.444022	2050.089171	444.732634	145.981978	19.599769	254.215401	350.660830	72.920827	2360.392784	117.730099	90
1020	3.280604	1972.372865	416.836524	110.533477	23.957502	248.423047	740.142791	74.734344	2662.906040	236.606764	90
1021	3.705351	2066.799773	741.475517	141.397963	19.246945	275.779840	641.468152	74.042708	2071.715856	197.126067	90
1022	3.808020	1890.413468	417.316232	129.183416	27.474763	300.952708	758.747882	74.309704	2856.328932	194.754342	90

999 rows × 13 columns

Рисунок 35 – Удаление выбросов

В соответствии с рисунком 35, видно, что количество строк в датасете уменьшилась на 24 замера, что составляет 2,35 процентов от исходной выборки.

Проверим корреляцию признаков. Чтобы посмотреть, как изменились зависимости, после удаления выбросов.

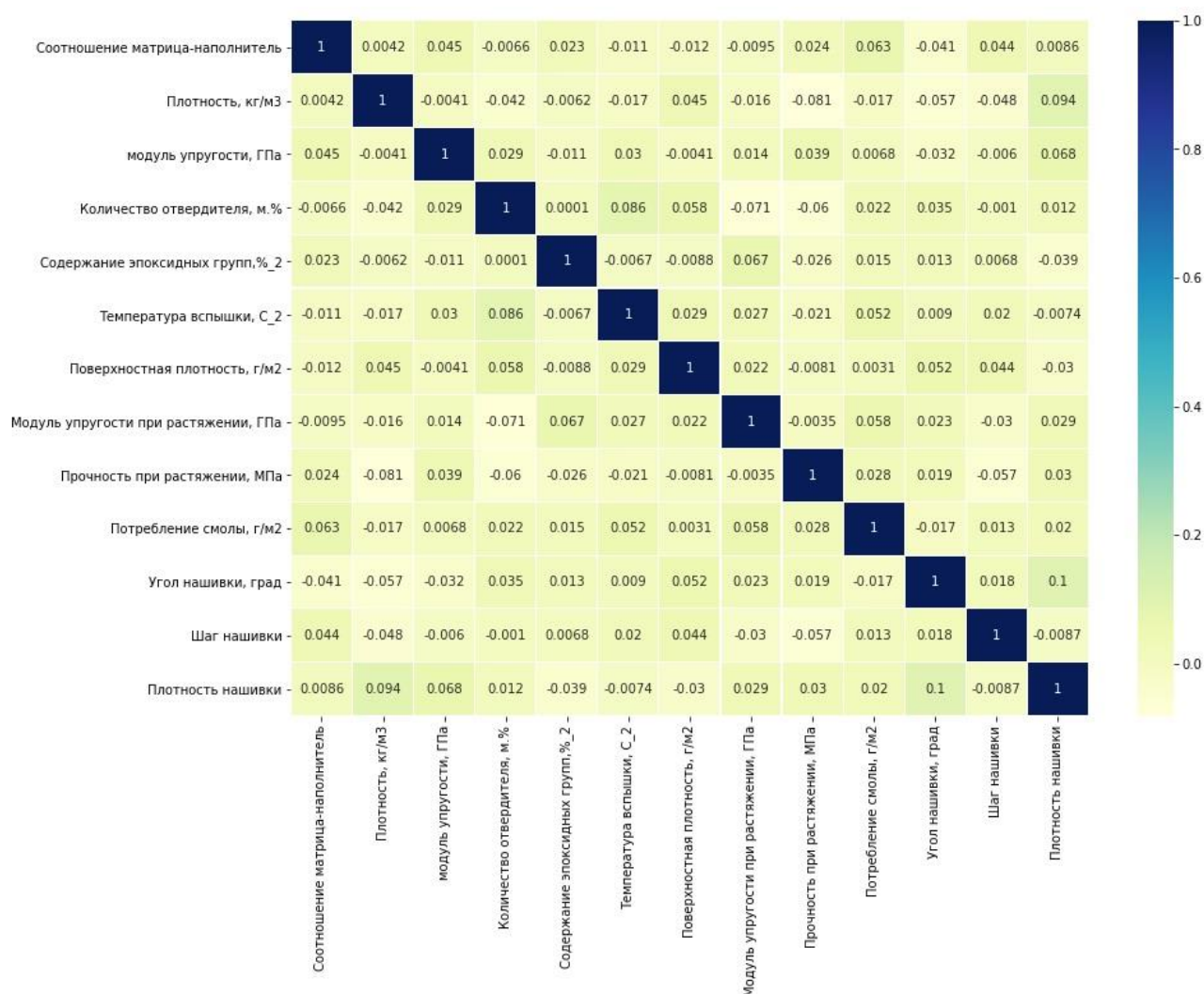


Рисунок 36 - Тепловая карта матрицы корреляции с удаленными выбросами

В соответствии с рисунком 36, видно, что в результате удаления выбросов, корреляция изменилась не значительно. Где-то возросла, где-то уменьшилась. Но существенных изменений нет, корреляция между признаками по-прежнему, фактически отсутствует.



Очистив, дата сет от выбросов, построим график распределения плотности ядра, для оценки необходимости нормализации.

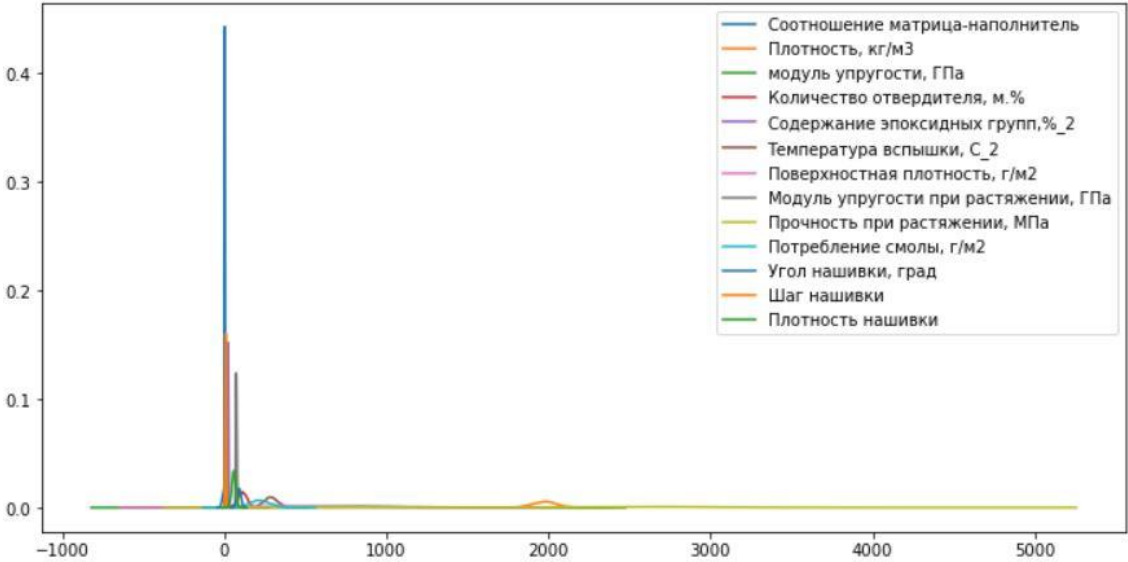


Рисунок 37 – График распределения плотности ядра

В соответствии с рисунком 37 наглядно видно, что данные находятся в очень разных диапазонах. Так как диапазоны очень разные, необходимо провести нормализацию данных.

Проведём нормализацию данных в соответствии с рисунком 38.

	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, град	
0	0.000499	0.545436	0.198490	0.013434	0.006381	0.076473	0.056424	0.018808	0.806064	0.059111	0.000000	(
1	0.000499	0.545011	0.198335	0.034634	0.005705	0.080543	0.056380	0.018793	0.805435	0.059065	0.000000	(
2	0.000744	0.544829	0.202097	0.030022	0.005976	0.076388	0.056362	0.018787	0.805167	0.059046	0.000000	(
3	0.000746	0.539271	0.201687	0.030161	0.006004	0.076742	0.056623	0.018874	0.808906	0.059320	0.000000	(
4	0.000699	0.519919	0.219673	0.030449	0.006062	0.077475	0.057164	0.019055	0.816627	0.059886	0.000000	(
...	...	...	...	...	...	...	...	...	...	...	...	...
994	0.000700	0.601520	0.281289	0.026806	0.006201	0.100077	0.064463	0.022522	0.735625	0.038520	0.027733	(
995	0.001078	0.641541	0.139172	0.045683	0.006133	0.079552	0.109733	0.022819	0.738645	0.036842	0.028164	(
996	0.000953	0.572927	0.121081	0.032107	0.006959	0.072161	0.214994	0.021709	0.773510	0.068729	0.026143	(
997	0.001191	0.664389	0.238353	0.045454	0.006187	0.088652	0.206205	0.023802	0.665970	0.063368	0.028931	(
998	0.001071	0.531558	0.117343	0.036325	0.007726	0.084624	0.213349	0.020895	0.803159	0.054762	0.025307	(

999 rows × 13 columns

Рисунок 38 – График нормализации данных

Предобработку данных закончили. Удалили выбросы и нормализовали значения данных.

## 2.2 Разработка и тестирование моделей

В соответствии с поставленной задачей, нужно осуществить разработку и обучение моделей машинного обучения для двух выходных параметров: «Прочность при растяжении» и «Модуль упругости при растяжении». Для каждого признака построение моделей осуществляется отдельно. Разделение нормализованных данных на обучающую и тестовую выборки (в соотношении 70 на 30%, согласно поставленной задаче).

Для признака «Модуль упругости при растяжении» были разработаны и обучены следующие модели в соответствии с рисунком 39:

- модель на основе линейной регрессии (метод `LinearRegression`);
- модель случайный лес (метод `RandomForestRegressor()`);
- модель k ближайших соседей (метод `KNeighborsRegressor()`);
- модель Стохастический градиентный спуск (метод `SGDRegressor()`);

	R2	RMSE	MAE
<code>LinearRegression</code>	0.802514	-0.001182	-0.000947
<code>SGDRegressor</code>	0.373852	-0.002117	-0.001677
<code>KNeighborsRegressor</code>	0.693047	-0.001477	-0.001192
<code>RandomForestRegressor</code>	0.799389	-0.001189	-0.000945

Рисунок 39 – График оценки моделей

Как видно из таблицы оценки, лучше всего справилась линейная регрессия и случайный лес. Стохастический градиентный спуск справился, хуже всего.

По заданию у нас задача построить модели и найти лучшие гиперпараметры. Построим для некоторых моделей графики и поиск гиперпараметров.



Линейная регрессия:

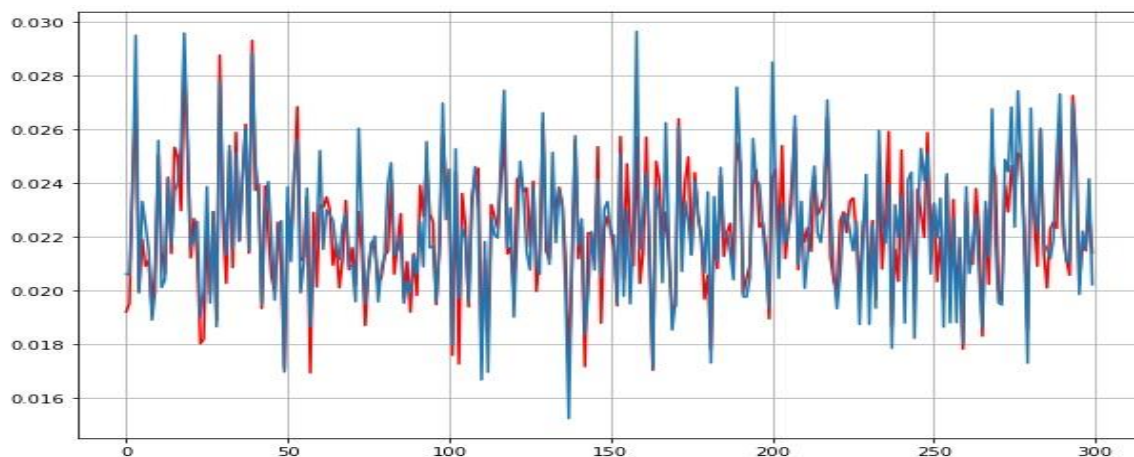


Рисунок 40 – Линейная модель модуля упругости при растяжении

Модель линейной регрессии справилась с задачей в 80,2 процентах случаев в соответствии с рисунком 40. Смогла выявить зависимость, но есть над чем поработать. Гиперпараметров у линейной регрессии нет, соответственно подбор оптимальных гиперпараметров не является возможным.

Случайный лес:

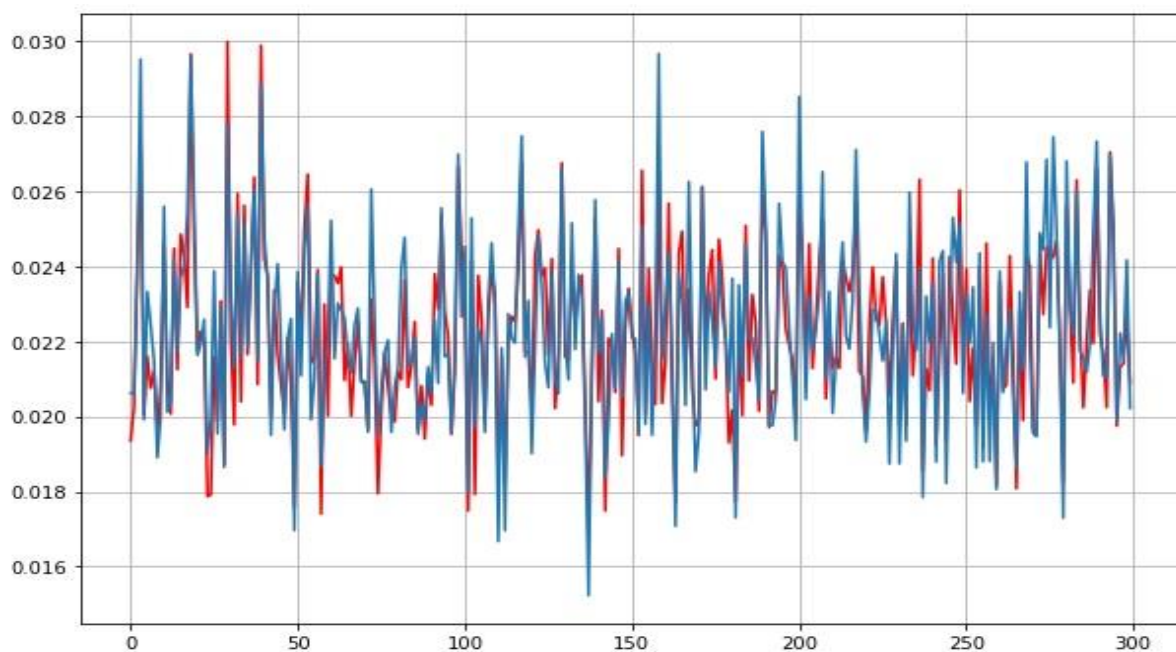


Рисунок 41 – Модель случайный лес

Случайный лес, в соответствии с рисунком 41, тоже справился с задачей и 79,9 процентов точность. Подбор оптимальных гиперпараметров модели

('bootstrap': True, 'max\_depth': 5, 'min\_samples\_leaf': 4, 'min\_samples\_split': 23, 'n\_estimators': 20). Результат чуть хуже линейной регрессии.

К-ближайших соседей:

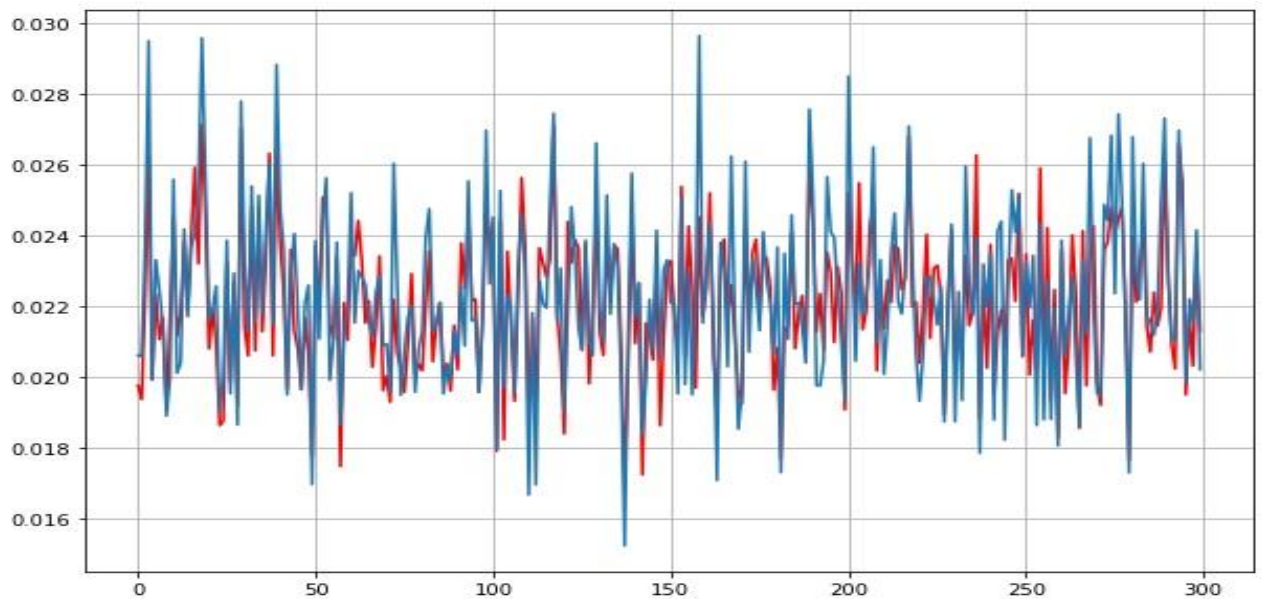


Рисунок 42 – Модель к-ближайших соседей

К -ближайших соседей справился с задачей плохо и смог выявить зависимость только в 69,9 процентах случаев. Подбор оптимальных гиперпараметров модели ('n\_neighbors': 13). Точность модели хуже линейной регрессии и случайного леса.

Стохастический градиентный спуск:

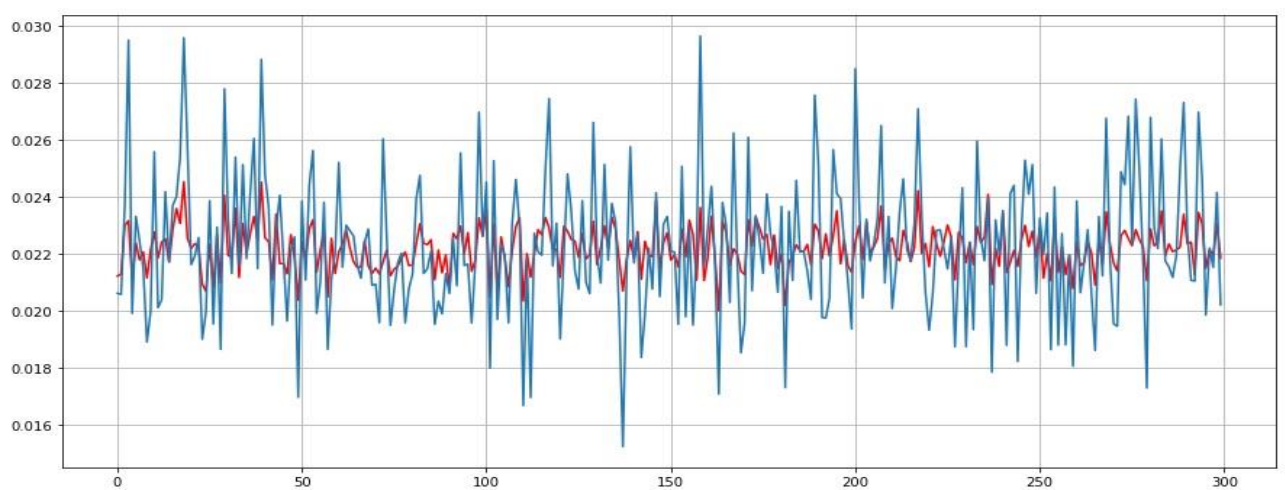


Рисунок 43 – Модель стохастический градиентный спуск

Стохастический градиентный спуск не справился с задачей и не смог выявить зависимость. Точность составила 37,4 процента. Подбор оптимальных гиперпараметров модели ('loss': 'squared\_epsilon\_insensitive', 'penalty': 'l2'). Ее точность осталась хуже остальных моделей.

С предсказанием модуля упругости лучше всего справились линейная регрессия, случайный лес, стохастический градиентный спуск. Метод К ближайших соседей не справился с задачей.

Перейдем ко второму признаку «Прочность при растяжении».

Для признака «Прочность при растяжении» были разработаны и обучены следующие модели в соответствии с рисунком 44:

- модель на основе линейной регрессии (метод `LinearRegression`);
- модель дерева решений (метод `DecisionTreeRegressor()`);
- модель градиентный бустинг (метод `GradientBoostingRegressor ()`);

	R2	RMSE	MAE
LinearRegression	0.953757	-0.015283	-0.011781
DecisionTreeRegressor	0.891681	-0.023273	-0.015958
GradientBoostingRegressor	0.978285	-0.010245	-0.006701

Рисунок 44 – График оценки моделей

Как видно из таблицы оценки, все модели смогли выявить зависимости и предсказать прогноз прочности при растяжении. Лучшие результаты у градиентного бустинга. По заданию у нас задача у нас построить моделей и найти лучшие гиперпараметры. Построим для некоторых моделей графики и поиск гиперпараметров.

### Линейная регрессия:

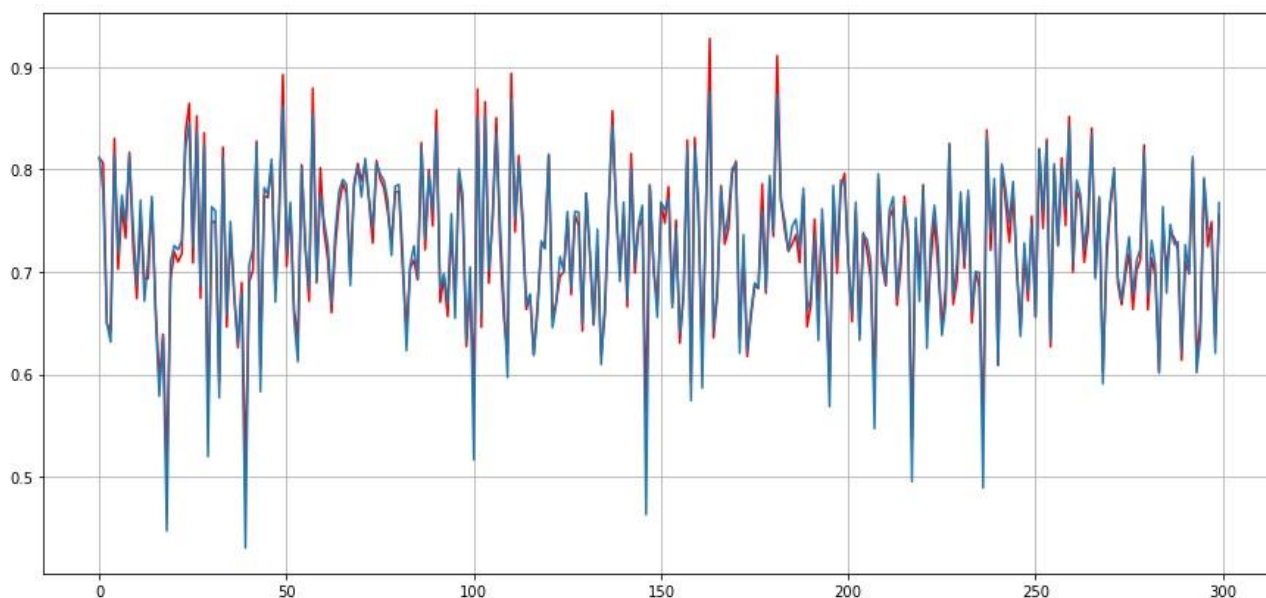


Рисунок 45 – Линейная модель прочности при растяжении

Модель линейной регрессии справилась с задачей в соответствии с рисунком 45. Смогла выявить зависимость в 95,4 процентах случаев. Гиперпараметров у линейной регрессии нет, соответственно подбор оптимальных гиперпараметров не является возможным.

### Деревья решений:

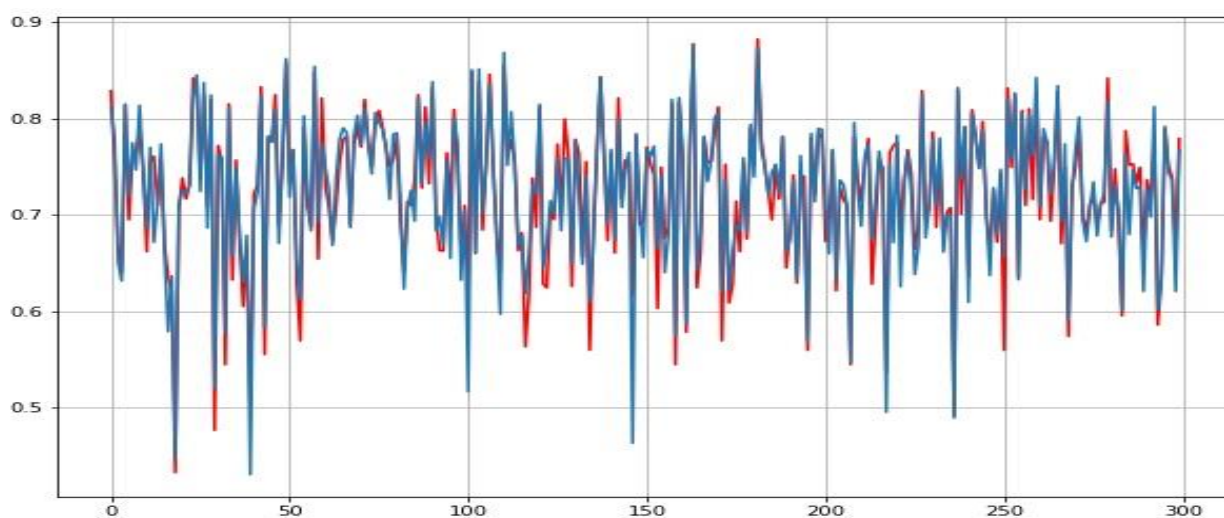


Рисунок 46 – Модель дерева решений

Деревья решений тоже справились с задачей и смогли выявить зависимость в 89,1 проценте. Подбор оптимальных гиперпараметров модели ('criterion':

'squared\_error', 'max\_depth': 3, 'max\_features': 'auto', 'min\_samples\_leaf': 100, 'min\_samples\_split': 200, 'splitter': 'best').

Градиентный

бустинг:

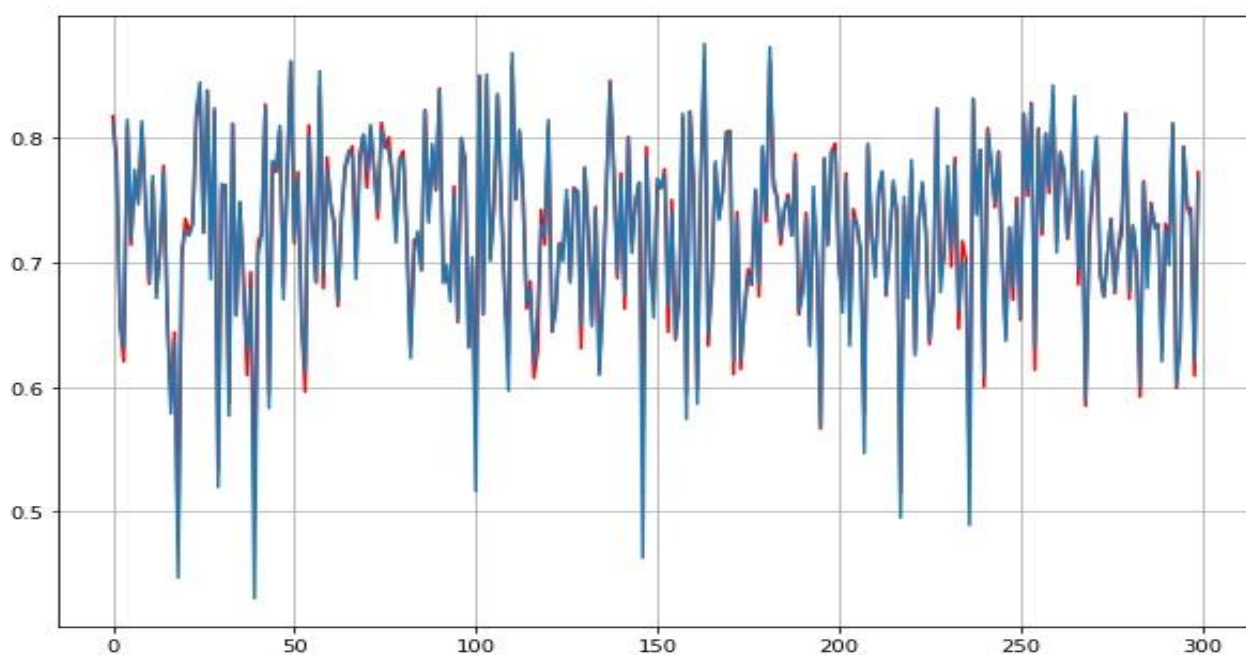


Рисунок 46 – Модель градиентный бустинг

Градиентный бустинг справился с задачей лучше всех и смог выявить зависимость с точностью 97,8 процентов.. Подбор оптимальных гиперпараметров модели ('criterion': 'mae', 'loss': 'absolute\_error', 'max\_depth': 2, 'min\_samples\_split': 7, 'n\_estimators': 10).

С задачей нахождения Прочности при растяжении все модели справились хорошо, лучшего всего справился градинтный бустинг.

## 2.3 Нейронная сеть, рекомендации соотношения матрица-наполнитель

При использовании библиотеки Keras (библиотека глубокого обучения) разработана простая модель для рекомендации соотношения «матрица-наполнитель».

Описание для архитектуры нейронной сети:

В модели используется четыре скрытых уровня. В первом уровне находятся 64 нейрона. В других скрытых уровнях находятся 64, 64 и 1 нейрона. Происходит сжатие информации - при снижении числа нейронов, которую сеть обрабатала на предыдущих уровнях.

Для данного эксперимента был выбран ReLU (функция активации - выпрямленная линейная единица).

После проведения обучения модели для была выполнена оценка модели, которая составила 0.0008667171932756901. Результат возможно мог бы быть и лучше.

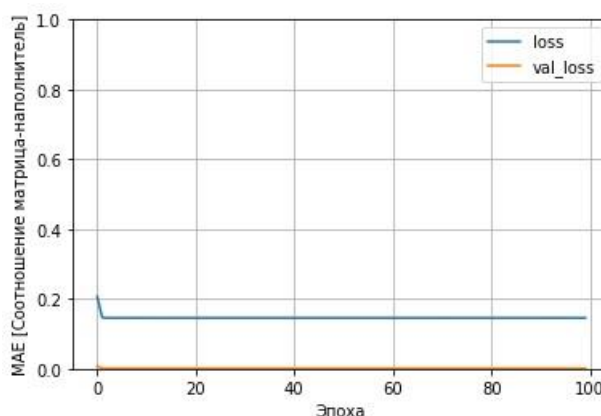


Рисунок 47 – Оценка нейронной сети





### **3. Заключение**

Теоретически разработанный метод определения надёжности изделий из композиционных материалов, основанный на использовании статистически достоверных характеристик материалов, полученных физическим и вычислительным экспериментом, позволяет оценивать уровень надёжности изделий как в отдельных точках, так и по всему объёму в целом.

Результаты работы выложены на GitHub:

[https://github.com/crtman/bmstu\\_final\\_qualifying\\_work](https://github.com/crtman/bmstu_final_qualifying_work)



#### **4. Список используемой литературы и ссылки на веб-ресурсы**

[1] К. Андерсон, Аналитическая культура. От сбора данных до бизнес-результатов: монография. Москва: O'Reilly, 2017, 392 с.

[2] How to choose a machine learning model in Python? – Режим доступа: <https://www.codeastar.com/choose-machine-learning-models-python/> (дата обращения 03.04.2022)

[3] Язык программирования Python: - Режим доступа: <https://www.python.org/> (дата обращения 01.04.2022)

[4] Библиотека Pandas – Режим доступа: <https://pandas.pydata.org/> (дата обращения 01.04.2022)

[5] Библиотека Matplotlib – Режим доступа: <https://matplotlib.org/> (дата обращения 01.04.2022)

[6] Библиотека Sklearn – Режим доступа: <https://scikit-learn.org/stable/> (дата обращения 01.04.2022)

[7] В.В. Васильев, В.Д. Протасов, В.В. Болотин и др.: Композитные материалы: справочник. Москва: Машиностроение, 1990, 510 с.

[8] Плас Дж. Вандер, Python для сложных задач: наука о данных и машинное обучение. Санкт-Петербург: Питер, 2018, 576 с.