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 Дорохова Ольга Васильевна

Москва, 2022

Введение

1. Аналитическая часть
   1. Постановка задачи
   2. Описание используемых методов
2. Практическая часть
   1. Предобработка данных
   2. Обучение моделей регрессии
   3. Тестирование модели
   4. Нейронная сеть для рекомендации соотношения матрица-наполнитель
   5. Разработка приложения
   6. Создание удаленного репозитория и выгрузка результатов работы

Заключение

Список использованной литературы

ВВЕДЕНИЕ

**Введение**

Композитные материалы (КМ) – это «супер-вещества», которые получают соединением двух и более компонентов. Компоненты в существенной степени отличаются друг от друга по своим свойствам. Сочетание их приводит к тому, что образуются новые материалы с уникальными свойствами, отличными от исходного сырья. То есть это суперновые материалы, вобравшие в себе, все лучшее от родителей.

Композитные соединения состоят из двух основных частей. Первая – это матрица, вторая – это наполнитель. Новые композитные вещества превосходят привычно используемые материалы по прочностным, механическим характеристиками и выгодно отличаются по массе, имея легкий вес.

Для примера композитных материалов можно привести клееную фанеру.

По своей структуре композитные материалы можно разделить на несколько групп:

* волокнистые;
* дисперсноупрочненные;
* упрочненные частицами;
* нанокомпозиты.

Волокнистые композиты – это материалы, улучшение свойств которых производится волокнами или нитевидными кристаллами. Например, кирпич с соломой. Незначительное введение добавки наполнителя приводит к появлению новых, уникальных свойств.

А, к примеру, добавка электропроводящих волокон придает материалу новое свойство – проводить электрический ток.

Если композитный материал имеет слоистую структуру, то в нем матрица и наполнитель располагаются слоями. Примером можно привести стекло, покрытое слоями полимерных пленок.

Другие представители композитных материалов имеют структуру, которая представлена матрицей и наполненными ее частицами, отличающимися по размерам.

* упрочненные материалы имеют 20-25% частиц, размер которых составляет более 1 мкм,
* дисперсноупрочненные – 1-15% частиц, размер которых составляет 0,01-0,1 мкм.
* нанокомпозитные материалы, имеют частицы, размер которых составляет – 10-100 нм.

Выделяют несколько видов композитных материалов:

* 1. Полимерные композитные материалы

Полимерные композитные материалы (ПКМ), имеют в качестве своей базы полимерную основу-матрицу. Это самый многочисленный вид КМ. Их применение позволило значительно снизить вес и улучшить эксплуатационные характеристики многих вещей. Так, к примеру, применение ПКМ при изготовлении искусственного спутника земли привело к снижению его веса, а облегчение на 1 кг дает экономию в 1000 долларов.

* 1. Стеклопластики

Полимерные композиты, армирование которых проводят стеклянными волокнами. Данные волокна получают формированием при нагревании неорганического стекла. В качестве матрицы выступают фенольные, эпоксидные смолы или термопластичные полимеры.

Материалы характеризуются: прозрачностью к радиоволнам, прочностью и электроизоляцией, низкой теплопроводностью.

Стеклопластики – это дешевый и доступный материал, его применяют в приборостроении, судостроении, строительстве, при изготовлении спортивных товаров.

### Углепластики

В данных композитных материалах наполнителями служат углеродсодержащие волокна, которые «добывают» из натуральных или искусственно созданных волокон.

Матрицей также служит термореактивный или термопластичный полимер.

Преимущества: низкая плотность, высокий коэффициент упругости, они легкие по массе, но в тоже время очень прочные, хорошо проводят электрический ток.

Применяются в авио-, машино- и ракетостроении, а также при производстве космической техники, спортивных товаров, медицинских протезов.

* 1. Боропластики

Матрица – термореактивный полимер.

Наполнитель – борные волокна, борные жгуты.

Борные волокна имеют большую прочность при сжатии, чем волокна других материалов. Поэтому и получаемые с их использованием материалы обладают отличными прочностными, износостойкими характеристиками, а также характеризуются инертностью к агрессивным средам. Но в тоже время, им свойственна хрупкость, что вводит определенные ограничения на использование.

Есть у данных ПКМ еще один минус – высокая цена, по данной причине область их использования достаточно узкая. Применяются главным образом только при изготовлении деталей, которые используются при высокой нагрузке и требует высокой надежности.

* 1. Органопластики

Основа – преимущественно используют эпоксидные, полимерные, фенольные смолы.

Наполнитель – искусственные или натуральные волокна. Волокна используются в виде нитей или жгутов, а также «полотен» - ткани, бумаги. Объем наполнителя составляет 40-70%.

Отличаются низкой плотностью, легкостью. Имеют высокую прочность. Отличное сопротивление к ударам, а также устойчивость к динамическому воздействию, нагрузкам. Но при всем при этом их прочность при изгибе и сжатии мала.

Применяются при машино-, авио-, судостроении, космической и авиационной технике, производстве спорт инвентаря и радиоэлектроники.

* 1. Полимеры наполненные порошками

В наше время применяют дешевые наполнители, такие как каолин, для термопластичной/термополимерной матрицы. Данные соединения используют для изготовления электроизоляционных материалов, труб. Сажу применяют для наполнения при изготовлении резин.

* 1. Текстолиты

Это полимерные материалы, имеющие слоистую структуру и применяемые для изготовления, например кухонных поверхностей, в качестве армирующего элемента используют ткани.

Композиционные материалы – это улучшенные соединения, это как валенки с калошами и тепло, и ноги не промокают.

Два ключевых момента, которые нужно знать о чудо композитных материалах:

* Материалы состоят из матрицы и наполнителя,
* Материалы после «модификации» приобретает новые свойства.

Существует определенный 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).

**1 Аналитическая часть**

* 1. **Постановка задачи.**

На основании имеющихся данных необходимо спрогнозировать следующие конечные свойства получаемых композиционных материалов:

* модуль упругости при растяжении;
* прочность при растяжении;
* соотношение матрица-наполнитель.

На входе имеются два датасета с данными о свойствах компонентов композиционных материалов. Основные сведения о данных датасетах представлены в Таблице 1 и Таблице 2.

Таблица 1 – Основные сведения датасета X\_bp.xlsx

# 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

Таблица 2 – Основные сведения датасета X\_nup.xlsx

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 Угол нашивки, град 1040 non-null float64

1 Шаг нашивки 1040 non-null float64

2 Плотность нашивки 1040 non-null float64

Для удобства использования датасеты объединяются по индексу тип объединения INNER с отсечением лишних строк из датасета X\_nup.xlsx.

Основная информация о полученном датасете представлена в Таблице 3.

Таблица 3 – Основные сведения объединенного датасета

# 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 float64

11 Шаг нашивки 1023 non-null float64

12 Плотность нашивки 1023 non-null float64

Данный датасет занимает 111.9 Kb.

* 1. **Описание используемых методов**

Для прогнозирования модуля упругости при растяжении и прочности при растяжении используем следующие методы машинного обучения:

* Линейная регрессия (множественная)

Множественная линейная регрессия (MLR), также известная просто как множественная регрессия, – это статистический метод, который использует несколько независимых переменных для прогнозирования результата переменной ответа. Цель множественной линейной регрессии – смоделировать линейную связь между независимыми переменными и ответной (зависимой) переменной. Множественная линейная регрессия вычисляется по формуле (1).

https://nesrakonk.ru/wp-content/uploads/2021/06/2021-12-16_09-12-35.png (1)

где для i - n наблюдений:

yi - зависимая переменная;

xi - объясняющие переменные;

β0 - y-интерцепт (постоянный член);

βp - коэффициенты наклона для каждой объясняющей переменной;

ϵ - член ошибки модели (также известный как остатки).

Простая линейная регрессия – это функция, которая позволяет аналитику или статистику делать прогнозы относительно одной переменной на основе информации, которая известна о другой переменной. Линейная регрессия может быть использована только при наличии двух непрерывных переменных – независимой переменной и зависимой переменной. Независимая переменная – это параметр, который используется для расчета зависимой переменной или результата. Модель множественной регрессии распространяется на несколько объясняющих переменных.

* Случайный лес решений

Это алгоритм машинного обучения, заключающийся в использовании ансамбля решающих деревьев. Алгоритм сочетает в себе две основные идеи: метод бэггинга Бреймана, и метод случайных подпространств предложенный Тин Кам Хо. Алгоритм применяется для задач классификации, регрессии и кластеризации.

Основная идея заключается в использовании большого ансамбля решающих деревьев, каждое из которых само по себе даёт очень невысокое качество классификации, но за счёт их большого количества результат получается хорошим.

Сама структура дерева представляет собой «листья» и «ветки». На рёбрах («ветках») дерева решения записаны признаки, от которых зависит целевая функция, в «листьях» записаны значения целевой функции, а в остальных узлах — признаки, по которым различаются случаи. Чтобы классифицировать новый случай, надо спуститься по дереву до листа и выдать соответствующее значение.

Подобные деревья решений широко используются в интеллектуальном анализе данных. Цель состоит в том, чтобы создать модель, которая предсказывает значение целевой переменной на основе нескольких переменных на входе.

Каждый лист представляет собой значение целевой переменной, изменённой в ходе движения от корня по рёбрам дерева до листа. Каждый внутренний узел сопоставляется с одной из входных переменных.  
Дерево может быть также «изучено» разделением исходных наборов переменных на подмножества, основанные на проверке значений признаков. Это действие повторяется на каждом из полученных подмножеств. Рекурсия завершается тогда, когда подмножество в узле имеет те же значения целевой переменной, таким образом, оно не добавляет ценности для предсказаний.

Для прогнозирования соотношения матрица-наполнитель воспользуемся построением нейронной сети.

* 1. **Разведочный анализ данных**

.info()Показывает количество признаков и количество показателей по ним, их тип данные, размер занимаемой памяти.

.describe()оценивает столбцы с числовым типом данных (int или float) и возвращает статистические данные, которые дают представление о распределении значений.

.isnull и .notnull – проверка на пропуски. При создании датасета из файла многие пустые столбцы импортируются как нулевое значение, что впоследствии создает проблемы при работе с данными. isnull() и notnull() используются для проверки и управления значениями NULL в датасете.

Гистограммы распределения показывают распределение переменной. Каждый столбик — частотность в определенном интервале. Применяются для визуализации, как часто какие значения встречаются в наборе данных.

Диаграммы рассеяния (также точечная диаграмма, [англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) scatter plot) — математическая [диаграмма](https://ru.wikipedia.org/wiki/%D0%94%D0%B8%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B0), изображающая значения двух переменных в виде точек на [декартовой плоскости](https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D0%BA%D0%B0%D1%80%D1%82%D0%BE%D0%B2%D1%8B_%D0%BA%D0%BE%D0%BE%D1%80%D0%B4%D0%B8%D0%BD%D0%B0%D1%82%D1%8B). Могут использоваться и полярные координаты, особенно в случаях, когда одна из переменных представляет собой физическое направление или имеет циклический характер.

На диаграмме рассеяния каждому наблюдению (или элементарной единице набора данных) соответствует точка, координаты которой равны значениям двух каких-то параметров этого наблюдения. Если предполагается, что один из параметров зависит от другого, то обычно значения независимого параметра откладывается по горизонтальной оси, а значения зависимого — по вертикальной. Диаграммы рассеяния используются для демонстрации наличия или отсутствия [корреляции](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D1%80%D1%80%D0%B5%D0%BB%D1%8F%D1%86%D0%B8%D1%8F) между двумя переменными.

Ящики с усами – это такой вид диаграммы, который в удобной форме показывает медиану (или, если нужно, среднее), нижний и верхний квартили, минимальное и максимальное значение выборки и выбросы.

Корреляция - важнейший фактор, лежащий в основе анализа данных. Он сообщает, как переменные в наборе данных связаны друг с другом и как они перемещаются относительно друг друга. Значение корреляции колеблется от -1 до +1. 0 Корреляция означает, что две переменные не зависят друг от друга. Положительная корреляция указывает на то, что переменные движутся в одном направлении, а отрицательная корреляция указывает на противоположное.

Тепловая карта используется для графического представления значений матрицы с разными цветовыми оттенками для разных значений. Он очень четко визуализирует общую матрицу.

**2 Практическая часть**

**2.1 Предобработка данных**

Проверку на пропуски осуществляем с помощью .isnull().

Соотношение матрица-наполнитель 0

Плотность, кг/м3 0

модуль упругости, ГПа 0

Количество отвердителя, м.% 0

Содержание эпоксидных групп,%\_2 0

Температура вспышки, С\_2 0

Поверхностная плотность, г/м2 0

Модуль упругости при растяжении, ГПа 0

Прочность при растяжении, МПа 0

Потребление смолы, г/м2 0

Угол нашивки, град 0

Шаг нашивки 0

Плотность нашивки 0

dtype: int64

Пропусков нет.

Убеждаемся, что все ячейки в датасете заполнены.

Соотношение матрица-наполнитель 1023

Плотность, кг/м3 1023

модуль упругости, ГПа 1023

Количество отвердителя, м.% 1023

Содержание эпоксидных групп,%\_2 1023

Температура вспышки, С\_2 1023

Поверхностная плотность, г/м2 1023

Модуль упругости при растяжении, ГПа 1023

Прочность при растяжении, МПа 1023

Потребление смолы, г/м2 1023

Угол нашивки, град 1023

Шаг нашивки 1023

Плотность нашивки 1023

dtype: int64

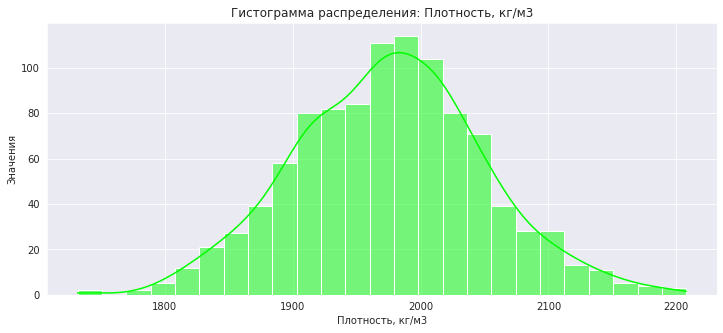
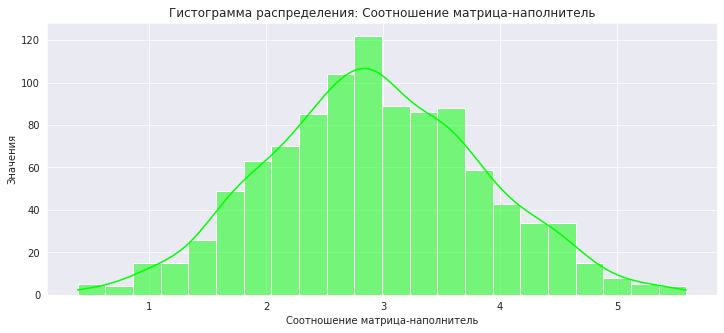
Общие сведения нашего датасета представлены в таблице 4.

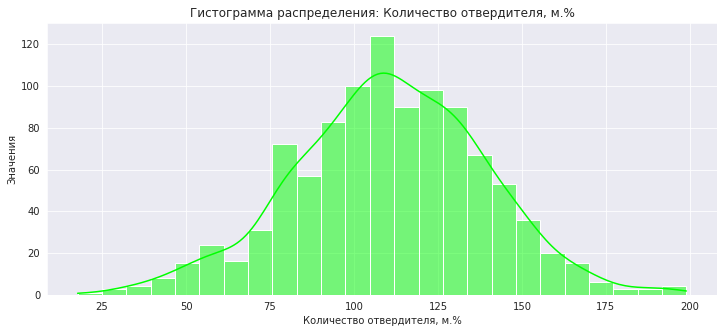
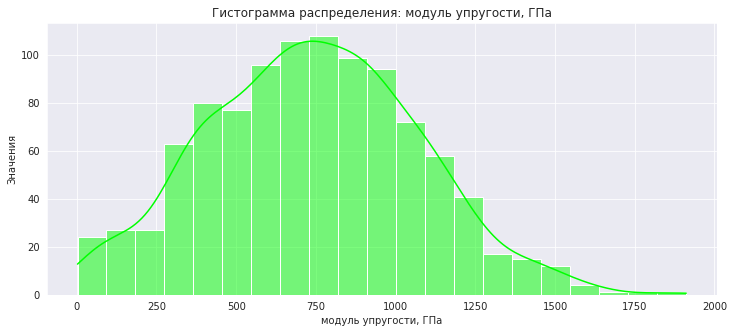
Таблица 4 - Общие сведения

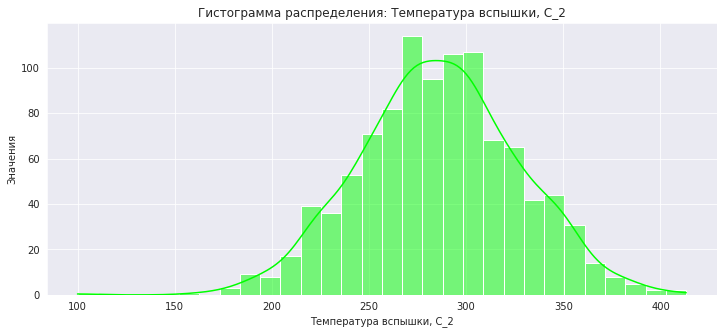
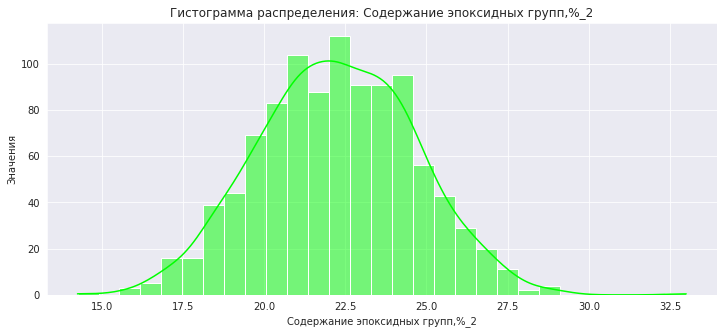
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **count** | **mean** | **std** | **min** | **25%** | **50%** | **75%** | **max** |
| **Соотношение матрица-наполнитель** | 1023.0 | 2.930366 | 0.913222 | 0.389403 | 2.317887 | 2.906878 | 3.552660 | 5.591742 |
| **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 |

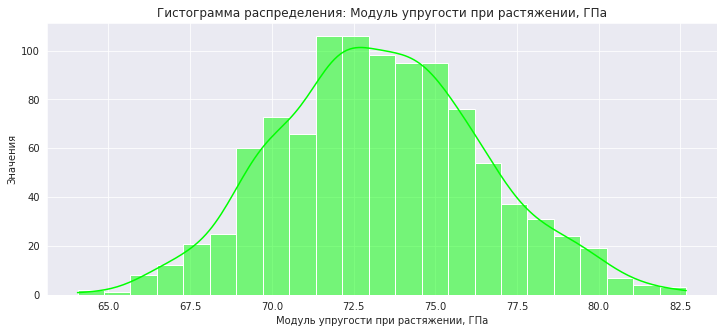
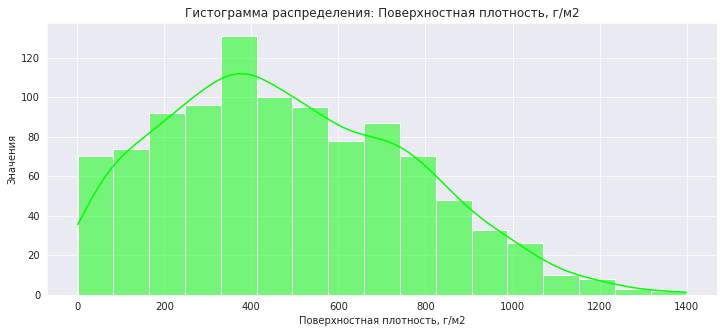
Выявляем дискретную величину «Угол нашивки, град».

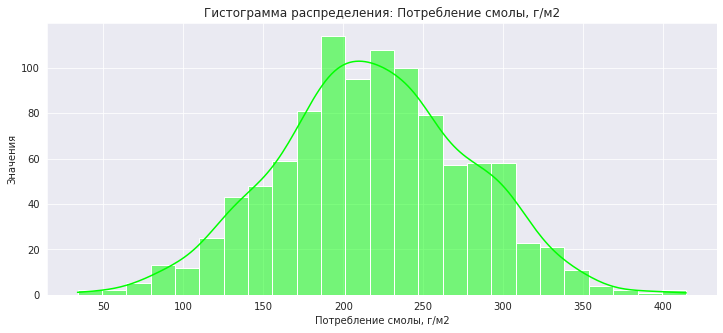
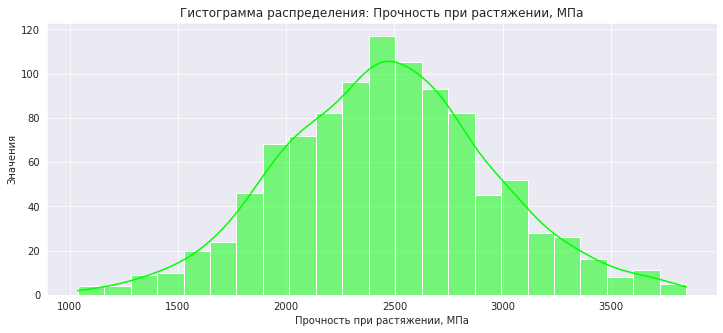
Строим гистограммы распределения (Рисунок 1).

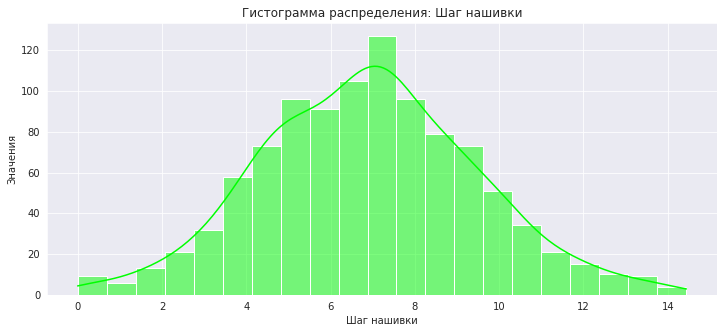
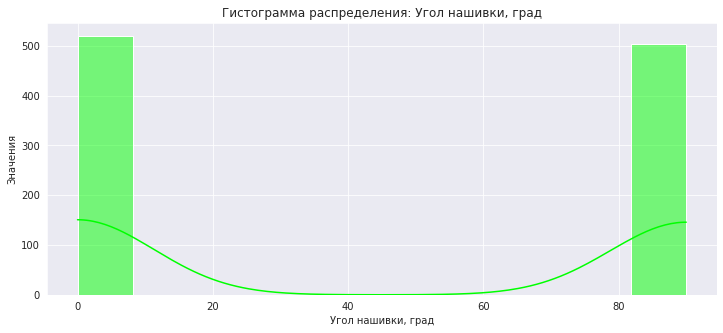












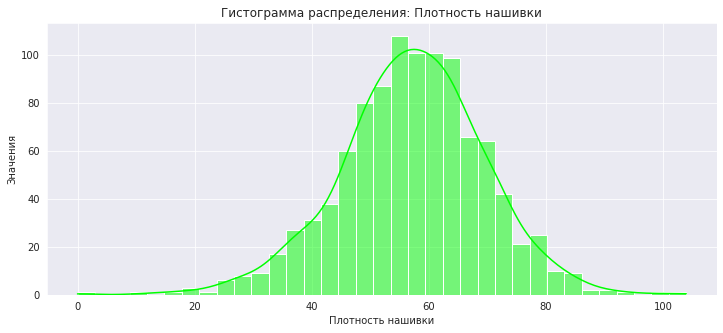


Рисунок 1 – Гистограммы распределения

Здесь уже наглядно убеждаемся в наличии дискретной величины «Угол нашивки, град». В основном все величины имеют нормальное распределение. Величина «Поверхностная плотность, г/м2» имеет распределение со смещением вправо.

Строим матрицу рассеивания (Рисунок 2), которая указывает нам об отсутствии линейной корреляции.

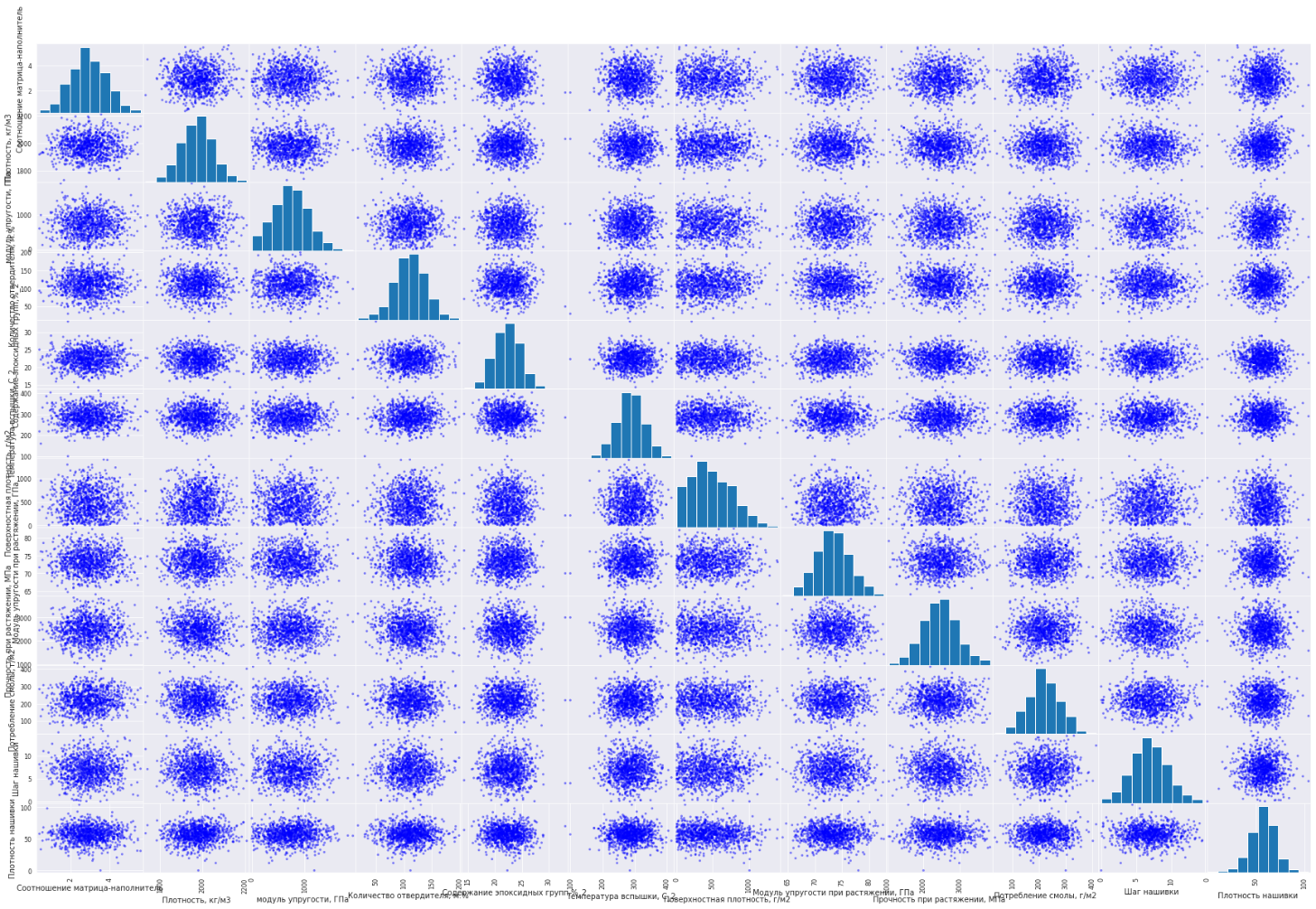


Рисунок 2 – Матричная диаграмма распределения

Тепловая карта корреляции (Рисунок 3) подтверждает отсутствие корреляции между признаками.

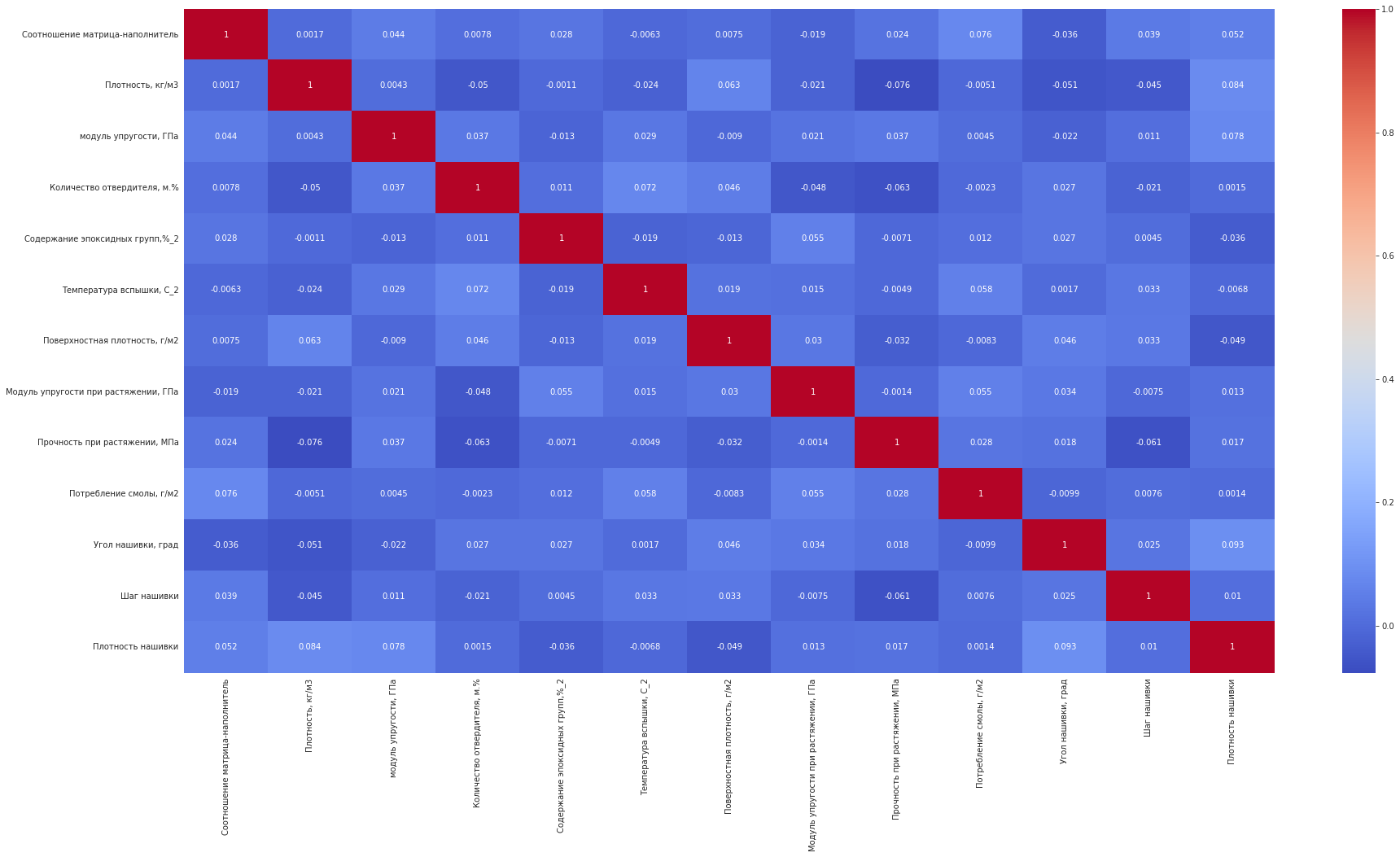
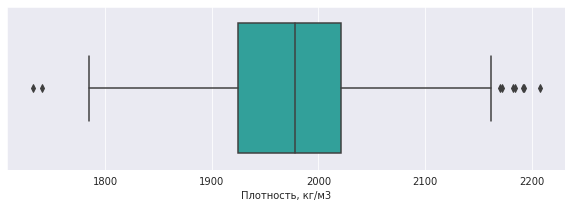
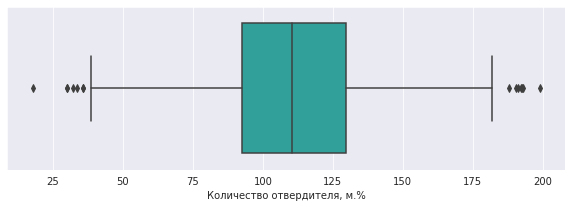
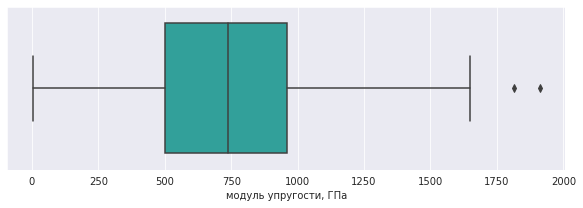
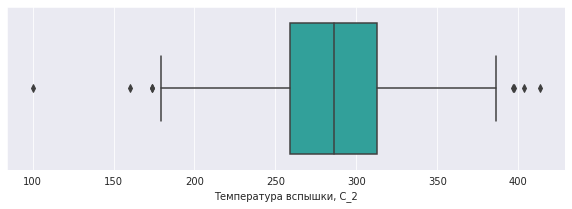
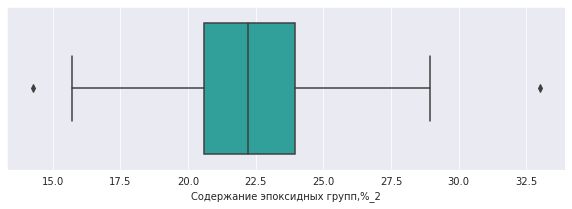


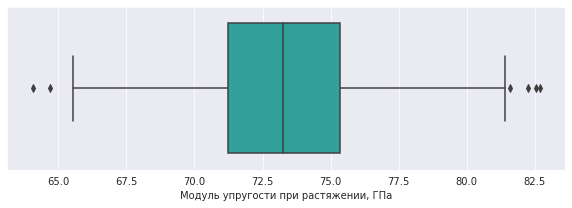
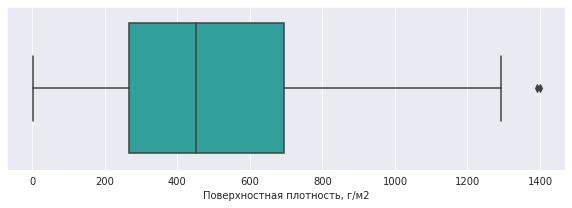
Рисунок 3 – Тепловая карта корреляции

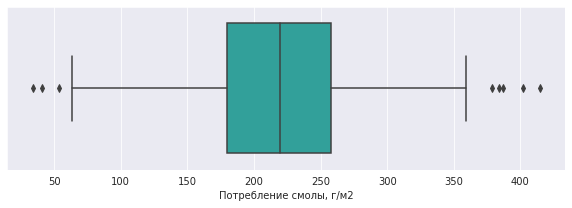
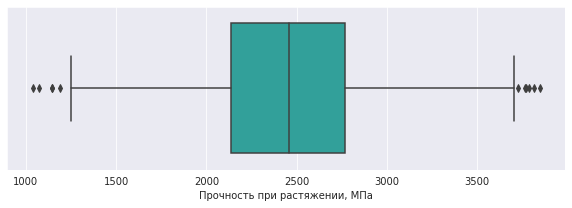
Строим ящики с усами (Рисунок 4) для всех признаков.

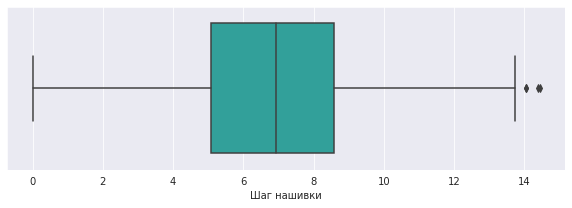
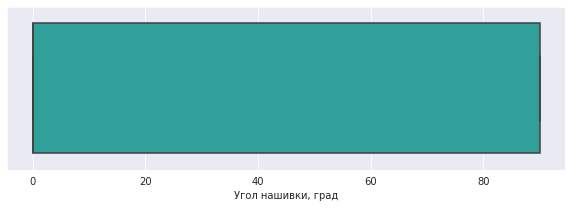












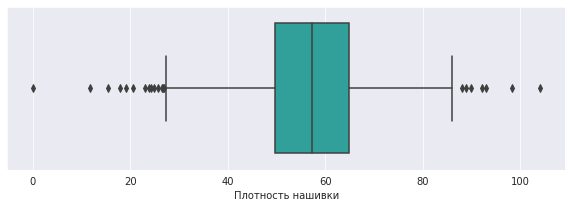


Рисунок 4 – Ящики с усами

Видим, что есть выбросы, от которых стоит избавиться

Избавляемся от выбросов методом межквантильного расстояния.

for p in data\_nule:

  q75, q25 = np.percentile(data\_nule.loc[:, p], [75, 25])

  iqr = q75 - q25

  max = q75 + (1.5 \* iqr)

  min = q25 - (1.5 \* iqr)

  data\_nule.loc[data\_nule[p] < min, p] = np.nan

  data\_nule.loc[data\_nule[p] > max, p] = np.nan

В результате избавляемся от 87 выбросов.

Соотношение матрица-наполнитель 6

Плотность, кг/м3 9

модуль упругости, ГПа 2

Количество отвердителя, м.% 14

Содержание эпоксидных групп,%\_2 2

Температура вспышки, С\_2 8

Поверхностная плотность, г/м2 2

Модуль упругости при растяжении, ГПа 6

Прочность при растяжении, МПа 11

Потребление смолы, г/м2 8

Угол нашивки, град 0

Шаг нашивки 4

Плотность нашивки 21

Проведем нормализацию данных с помощью метода MinMaxScaker() из библиотеки sklearn. После чего все значения признаков будут принадлежать диапазону от 0 до 1.

Основные параметры датасета после нормализации представлены в Таблице 5.

Таблица 5 – Общие сведения после нормализации

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

Посмотрим на ящики с усами (Рисунок 5) после нормализации.

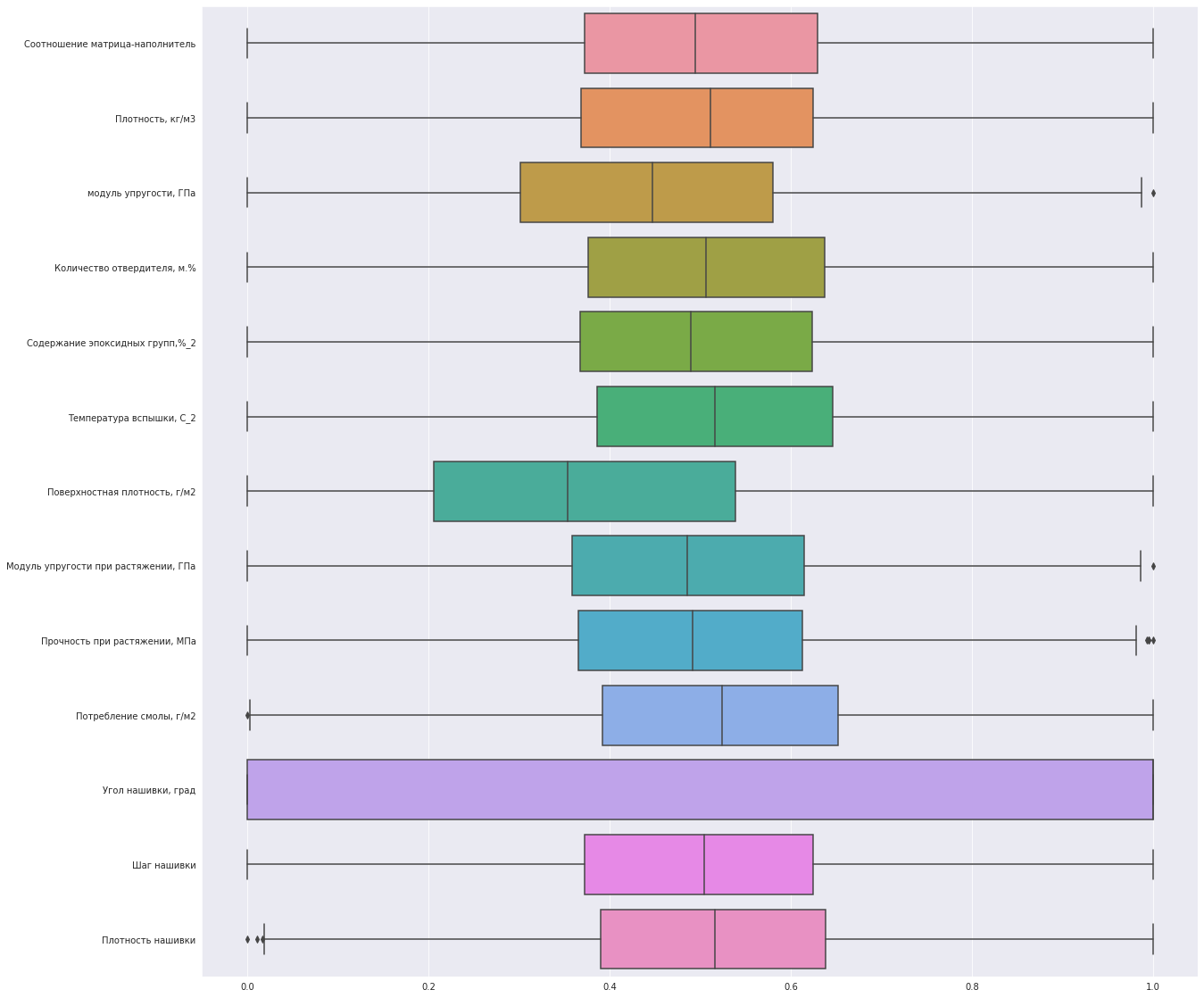


Рисунок 5 – Ящики с усами после нормализации

**2.2 Разработка и обучение модели**

Построение модели для модуля упругости при растяжении.

Линейная модель

from sklearn.linear\_model import LinearRegression

model\_LRypr = LinearRegression().fit(X\_tr\_ypr, y\_tr\_ypr)

y\_pred\_LRypr = model\_LRypr.predict(X\_test\_ypr)

R2\_LRypr = model\_LRypr.score(X\_test\_ypr, y\_test\_ypr)

MSE\_LRypr = mean\_squared\_error(y\_test\_ypr, y\_pred\_LRypr)

MAE\_LRypr = mean\_absolute\_error(y\_test\_ypr, y\_pred\_LRypr)

print(f'R2: {R2\_LRypr}\nMSE: {MSE\_LRypr}\nMAE: {MAE\_LRypr}')

Метрики

R2: -0.013477482035175736

MSE: 0.03779997307716396

MAE: 0.16027374246652173

RandomForestRegressor

from sklearn.ensemble import RandomForestRegressor

model\_RFRypr = RandomForestRegressor().fit(X\_tr\_ypr, y\_tr\_ypr)

y\_pred\_RFRypr = model\_RFRypr.predict(X\_test\_ypr)

R2\_RFRypr = model\_RFRypr.score(X\_test\_ypr, y\_test\_ypr)

MSE\_RFRypr = mean\_squared\_error(y\_test\_ypr, y\_pred\_RFRypr)

MAE\_RFRypr = mean\_absolute\_error(y\_test\_ypr, y\_pred\_RFRypr)

print(f'R2: {R2\_RFRypr}\nMSE: {MSE\_RFRypr}\nMAE: {MAE\_RFRypr}')

Метрики

R2: -0.042928577358004594

MSE: 0.03889841939691878

MAE: 0.16118764027759167

RandomForestRegressor + GSCV

from sklearn.model\_selection import GridSearchCV

param\_grid = {'n\_estimators': [30, 40, 50], 'max\_features': [1, 'sqrt', 'log2'], 'min\_samples\_leaf':[10, 20], 'min\_samples\_split': [2, 4]}

model\_RFypr = RandomForestRegressor()

GSCVypr = GridSearchCV(model\_RFypr, param\_grid, cv=10)

GSCVypr.fit(X\_tr\_ypr, y\_tr\_ypr)

y\_pred\_GSCVypr = GSCVypr.predict(X\_test\_ypr)

R2\_GSCVypr = GSCVypr.score(X\_test\_ypr, y\_test\_ypr)

MSE\_GSCVypr = mean\_squared\_error(y\_test\_ypr, y\_pred\_GSCVypr)

MAE\_GSCVypr = mean\_absolute\_error(y\_test\_ypr, y\_pred\_GSCVypr)

print(f'R2: {R2\_GSCVypr}\nMSE: {MSE\_GSCVypr}\nMAE: {MAE\_GSCVypr}')

Метрики

R2: 0.003605028340145644

MSE: 0.03716284157328402

MAE: 0.15855524724180167

Лучшее решение:

{'max\_features': 1,

'min\_samples\_leaf': 20,

'min\_samples\_split': 2,

'n\_estimators': 40}

Построение модели для прочности при растяжении

Линейная модель

model\_LRpr = LinearRegression().fit(X\_tr\_pr, y\_tr\_pr)

y\_pred\_LRpr = model\_LRpr.predict(X\_test\_pr)

R2\_LRpr = model\_LRpr.score(X\_test\_pr, y\_test\_pr)

MSE\_LRpr = mean\_squared\_error(y\_test\_pr, y\_pred\_LRpr)

MAE\_LRpr = mean\_absolute\_error(y\_test\_pr, y\_pred\_LRpr)

print(f'R2: {R2\_LRpr}\nMSE: {MSE\_LRpr}\nMAE: {MAE\_LRpr}')

Метрики

R2: -0.02660478122166854

MSE: 0.03209343003645218

MAE: 0.1421350665750559

RandomForestRegressor

model\_RFRpr = RandomForestRegressor().fit(X\_tr\_pr, y\_tr\_pr)

y\_pred\_RFRpr = model\_RFRpr.predict(X\_test\_pr)

R2\_RFRpr = model\_RFRpr.score(X\_test\_pr, y\_test\_pr)

MSE\_RFRpr = mean\_squared\_error(y\_test\_pr, y\_pred\_RFRpr)

MAE\_RFRpr = mean\_absolute\_error(y\_test\_pr, y\_pred\_RFRpr)

print(f'R2: {R2\_RFRpr}\nMSE: {MSE\_RFRpr}\nMAE: {MAE\_RFRpr}')

Метрики

R2: -0.05782129672057357

MSE: 0.03306931196732896

MAE: 0.14466756205307238

RandomForestRegressor + GSCV

param\_grid = {'n\_estimators': [30, 40, 50], 'max\_features': [1, 'sqrt', 'log2'], 'min\_samples\_leaf':[10, 20], 'min\_samples\_split': [2, 4]}

model\_RFpr = RandomForestRegressor()

GSCVpr = GridSearchCV(model\_RFpr, param\_grid, cv=10)

GSCVpr.fit(X\_tr\_pr, y\_tr\_pr)

y\_pred\_GSCVpr = GSCVpr.predict(X\_test\_pr)

R2\_GSCVpr = GSCVpr.score(X\_test\_pr, y\_test\_pr)

MSE\_GSCVpr = mean\_squared\_error(y\_test\_pr, y\_pred\_GSCVpr)

MAE\_GSCVpr = mean\_absolute\_error(y\_test\_pr, y\_pred\_GSCVpr)

print(f'R2: {R2\_GSCVpr}\nMSE: {MSE\_GSCVpr}\nMAE: {MAE\_GSCVpr}')

Метрики

R2: -0.0012347658836766584

MSE: 0.03130031974983863

MAE: 0.14068170130558652

Лучшее решение

{'max\_features': 1,

'min\_samples\_leaf': 20,

'min\_samples\_split': 4,

'n\_estimators': 30}

**2.3 Тестирование модели**

Сравнительный анализ моделей по метрикам представлен в Таблице 6.

Таблица 6 – Сравнительный анализ моделей

|  |  |  |  |
| --- | --- | --- | --- |
| **Модель регрессии** | **R2** | **MSE** | **MAE** |
| Линейная регрессия МУ | -0.013477 | 0.037800 | 0.160274 |
| Случайный лес МУ | -0.042929 | 0.038898 | 0.161188 |
| Случайный лес + GSCV МУ | 0.003605 | 0.037163 | 0.158555 |
| Линейная регрессия ПР | -0.026605 | 0.032093 | 0.142135 |
| Случайный лес ПР | -0.042929 | 0.033069 | 0.144668 |
| Случайный лес + GSCV ПР | -0.001235 | 0.031300 | 0.140682 |

**2.4 Нейронная сеть для рекомендации соотношения матрица-наполнитель**

Построим три нейронные сети для рекомендации соотношения матрица-наполнитель

Первая с одним слоем

linear\_model = tf.keras.Sequential([

    normalizer,

    layers.Dense(units=1)

])

linear\_model.compile(

    optimizer=tf.optimizers.Adam(learning\_rate=0.1),

    loss='mean\_absolute\_error')

Посмотрим график уменьшения ошибок (Рисунок 6)

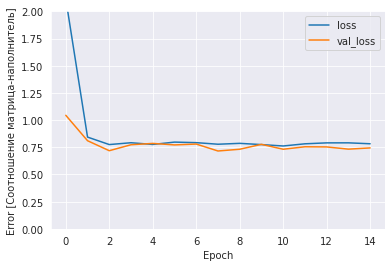


Рисунок 6 \_График уменьшения ошибки

Вторая модель – двухслойная.

DNN\_model = keras.Sequential([

      normalizer,

      layers.Dense(18, activation='relu'),

      layers.Dense(1)

    ])

DNN\_model.compile(loss='mean\_absolute\_error',

                optimizer=tf.keras.optimizers.Adam(0.001))

Посмотрим график уменьшения ошибок этой модели (Рисунок 7)

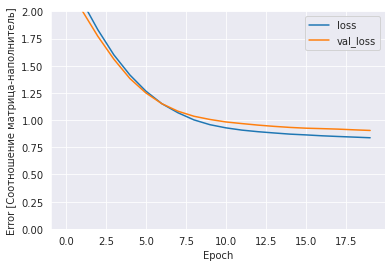


Рисунок 7 \_График уменьшения ошибки

Третья модель – нейронная сеть с тремя слоями

DNNN\_model = keras.Sequential([

      normalizer,

      layers.Dense(18, activation='relu'),

      layers.Dense(18, activation='relu'),

      layers.Dense(1)

    ])

DNNN\_model.compile(loss='mean\_absolute\_error',

                optimizer=tf.keras.optimizers.Adam(0.001))

Посмотрим график уменьшения ошибок этой модели (Рисунок 8)

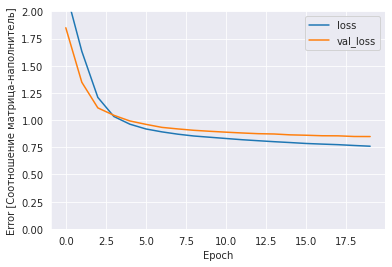


Рисунок 8 \_График уменьшения ошибки

Сравнительный анализ моделей по метрикам представлен в Таблице 7

Таблица 7 – Сравнительный анализ моделей

|  |  |  |
| --- | --- | --- |
|  | Mean absolute error | R2 |
| DNNN\_model | 0.8485245108604431 | -0.14498861122268036 |
| DNN\_model | 0.9046271443367004 | -0.6027963424490668 |
| linear\_model | 0.7439369559288025 | -0.4213725228206462 |

**2.5 Разработка приложения**

Приложение было разработано для предсказания модуля упругости при растяжении с помощью FLASK. Интерфейс приложения можно посмотреть на рисунке 9.

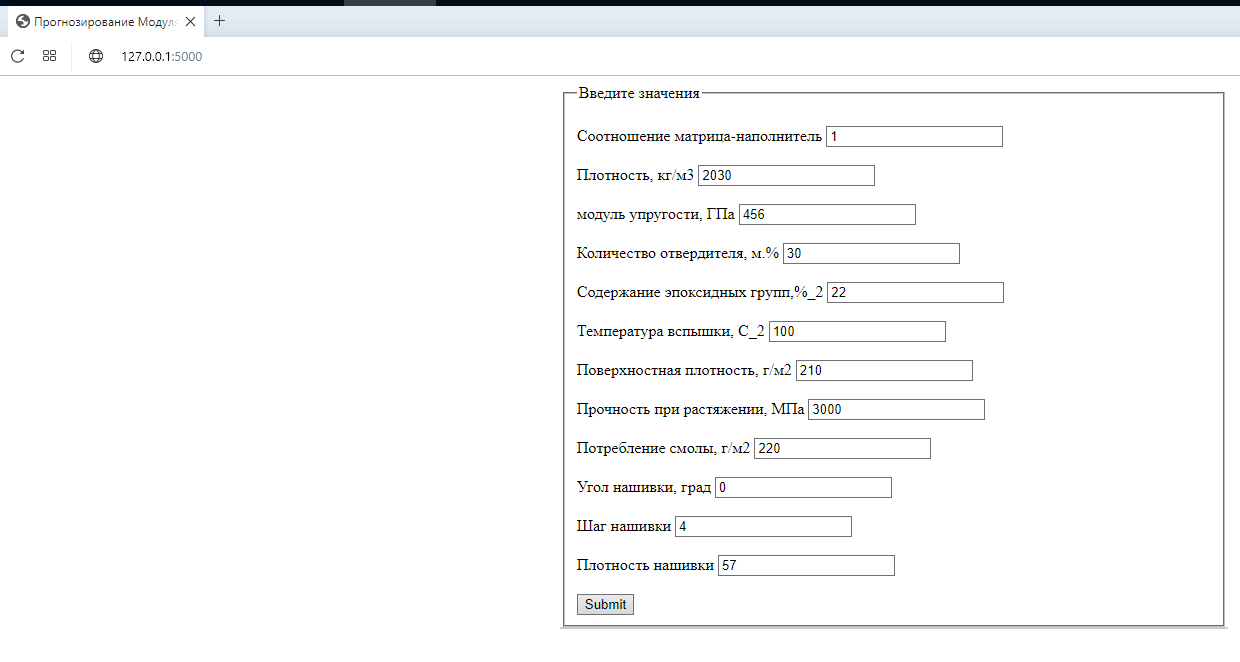


Рисунок 9 – Приложение\_График уменьшения ошибки

Код приложения доступен в репозитории на GitHab, все данные которого представлены в следующей главе.

**2.6 Создание удаленного репозитория и загрузка результатов работы на него.**

Указывается страница слушателя, созданный репозиторий, коммиты в репозитории.

**Заключение**

После проведения анализа входных данных и построения моделей машинного обучения можно сделать вывод, что об искственном происхождении данныхрешениеПроанализировав входные данные и построив модели можно сделать вывод, что задача не имеет какого-либо удовлетворительного решения в связи с тем, что данные на входе представляют собой датасет, который:

**Список использованной литературы:**

1. Полимерные композиционные материалы на основе волокон различной химической природы / Ю.А. Кадыкова, А.Н. Леонтьев, О.Г. Васильева, С.Е. Артеменко // Строительные материалы, оборудование, технологии ХХI века. 2002. № 6. С. 10-11. 2
2. Сравнительные характеристики базальто-, стекло- и углепластиков, сформованных методом поликонденсационного наполнения / Ю.А. Кадыкова, О.Г. Васильева, С.Е. Артеменко, А.Н. Леонтьев // Пластические массы. 2003. № 5. С. 37-38
3. В.А.Головко. От многослойных персептронов к нейронным сетям глубокого доверия: парадигмы обучения и применение. В сб.: Нейроинформатика-2015. XVII Всероссийская научно-техническая конференция с международным участием. Лекции по нейроинформатике, с.47-84. НИЯУ МИФИ, 2015
4. Перевод: А.Мюллер, С.Гвидо. Введение в машинное обучение с помощью Python. Руководство для специалистов по работе с данными. Вильямс, 2017, 480 c.
5. Перевод: О.Жерон. Прикладное машинное обучение с помощью Scikit-Learn и TensorFlow. Концепции, инструменты и техники для создания интеллектуальных систем. Вильямс, 2018, 688 c.
6. Виды и применение композитных материалов. Режим доступа: https://epolymer.ru/article/vidy-i-primenenie-kompozitnyh-materialov. (Дата обращения: 15.06.2022).
7. Random\_forest. Режим доступа: https://ru.wikipedia.org/wiki/Random\_forest. (Дата обращения: 15.06.2022).