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 Кондрашина Татьяна Александровна

Москва, 2022

**Содержание**

[**Введение** 3](#_Toc118928097)

[**1.** **Аналитическая часть** 4](#_Toc118928098)

[**1.1.** **Постановка задачи** 4](#_Toc118928099)

[**1.2.** **Описание используемых методов** 5](#_Toc118928100)

[**1.3.** **Разведочный анализ данных** 9](#_Toc118928101)

[**2.** **Практическая часть** 19](#_Toc118928102)

[**2.1.** **Предобработка данных** 19](#_Toc118928103)

[**2.2.** **Разработка и обучение модели** 23](#_Toc118928104)

[**2.3.** **Тестирование модели** 23](#_Toc118928105)

[**2.4.** **Создание нейронной сети** 25](#_Toc118928106)

[**2.5.** **Разработка приложения** 27](#_Toc118928107)

[**2.6.** **Создание удаленного репозитория и загрузка результатов работы на него** 27](#_Toc118928108)

[**Заключение** 28](#_Toc118928109)

[**Список использованной литературы** 29](#_Toc118928110)

[**Приложение 1** 31](#_Toc118928111)

# **Введение**

Композиционные материалы – это искусственно созданные материалы, состоящие из нескольких других с четкой границей между ними. Композиты обладают теми свойствами, которые не наблюдаются у компонентов по отдельности. При этом композиты являются монолитным материалом, т. е. компоненты материала неотделимы друг от друга без разрушения конструкции в целом.

Яркий пример композита – железобетон. Бетон прекрасно сопротивляется сжатию, но плохо растяжению. Стальная арматура внутри бетона компенсирует его неспособность сопротивляться сжатию, формируя тем самым новые, уникальные свойства.

Современные композиты изготавливаются из других материалов: полимеры, керамика, стеклянные и углеродные волокна, но данный принцип сохраняется. У такого подхода есть и недостаток: даже если мы знаем характеристики исходных компонентов, определить характеристики композита, состоящего из этих компонентов, достаточно проблематично. Для решения этой проблемы есть два пути: физические испытания образцов материалов, или прогнозирование характеристик. Суть прогнозирования заключается в симуляции представительного элемента объема композита, на основе данных о характеристиках входящих компонентов (связующего и армирующего компонента).

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

## **Аналитическая часть**

### **Постановка задачи**

**На входе** имеются данные о начальных свойствах компонентов композиционных материалов (количество связующего, наполнителя, температурный режим отверждения и т.д.). **На выходе** необходимо спрогнозировать ряд конечных свойств получаемых композиционных материалов. Кейс основан на реальных производственных задачах Центра НТИ «Цифровое материаловедение: новые материалы и вещества» (структурное подразделение МГТУ им. Н.Э. Баумана).

Для работы даны 2 файла со свойствами композитов – <https://drive.google.com/file/d/1B1s5gBlvgU81H9GGolLQVw_SOi-vyNf2/view?usp=sharing>

Задачи:

1. провести разведочный анализ данных, нарисовать гистограммы распределения каждой из переменной, диаграммы «ящика с усами», попарные графики рассеяния точек;
2. для каждой колонки получить среднее, медианное значение, провести анализ и исключение выбросов, проверить наличие пропусков;
3. предобработать данные: удалить шумы и выбросы, сделать нормализацию и стандартизацию;
4. обучить несколько моделей для прогноза модуля упругости при растяжении и прочности при растяжении;
5. написать нейронную сеть, которая будет рекомендовать соотношение матрица-наполнитель;
6. разработать приложение с графическим интерфейсом, которое будет выдавать прогноз соотношения «матрица-наполнитель»; оценить точность модели на тренировочном и тестовом датасете;
7. создать репозиторий в GitHub и разместить код исследования, оформить файл README.

### **Описание используемых методов**

Рассматриваемая задача в рамках классификации категорий машинного принадлежит к задачам машинного обучения с учителем или к задаче регрессии. Цель любого алгоритма обучения с учителем — определить функцию потерь и минимизировать её. Для решения в процессе исследования были применены следующие методы:

* случайный лес;
* К-ближайших соседей;
* дерево решений.

Случайный лес (Random Forest) – это множество решающих деревьев. Универсальный алгоритм машинного обучения с учителем, представитель ансамблевых методов.  Если точность дерева решений оказалось 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 недообучаться;
* трудоёмко прогнозируемый.

Метод ближайших соседей – К-ближайших соседей (kNN – kNearest 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 (Decision Tree Regressor) – 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.

Используемые метрики качества моделей:

* средняя абсолютная ошибка MAE (Mean Absolute Error) рассчитывается как среднее абсолютных разностей между целевыми значением и значением, предсказанным моделью на данном обучающем примере в процессе обучения;
* средняя квадратическая ошибка MSE (Mean Squared Error) рассчитывается как среднее арифметическое квадратов разностей между предсказанными и реальными значениями, Чем ближе к нулю MSE, тем лучше работают предсказательные качества модели;
* коэффициент детерминации R2 измеряет долю дисперсии, объяснённую моделью, в общей дисперсии целевой переменной. Если он близок к единице, то модель хорошо объясняет данные, если же он близок к нулю, то качество прогноза идентично средней величине целевой переменной (т.е. очень низкое). Отрицательные значение коэффициента детерминации означают плохую объясняющую способность модели;
* средняя абсолютная процентная ошибка MAPE (Mean Absolute Percentage Error) – средняя абсолютная ошибка, выраженная в процентах, показывает процент ошибок модели.

### **Разведочный анализ данных**

С целью получения первоначальных представлений о характерах распределений переменных исходного набора данных, формирование оценки качества исходных данных (наличие пропусков, выбросов), выявление характера взаимосвязи между переменными проводим разведочный анализ данных.

Для данной работы были даны 2 файла: X\_bp.xlsx, состоящий из 1023 строки и 11 столбцов и X\_nup.xlsx , состоящий из 1040 строк и 4 столбцов.

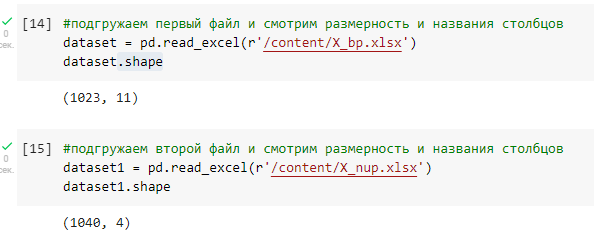


Рисунок 1 – Начало работы с файлами

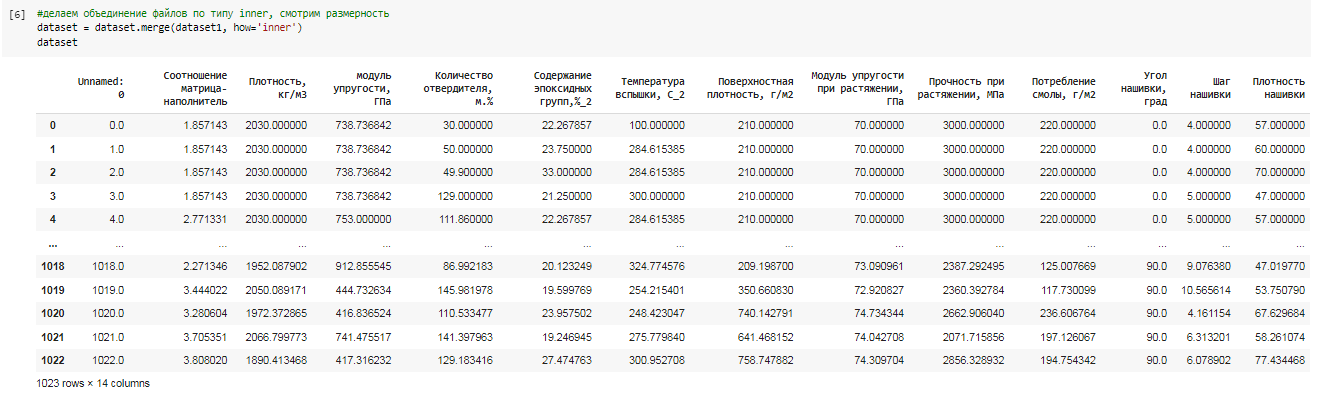


Рисунок 2 – Объединение файлов по типу INNER

При анализе полученного датасета видим неинформативность первого столбца, принимаем решение о его удалении. Размер полученного датасета – 1023 строки, 13 столбцов.

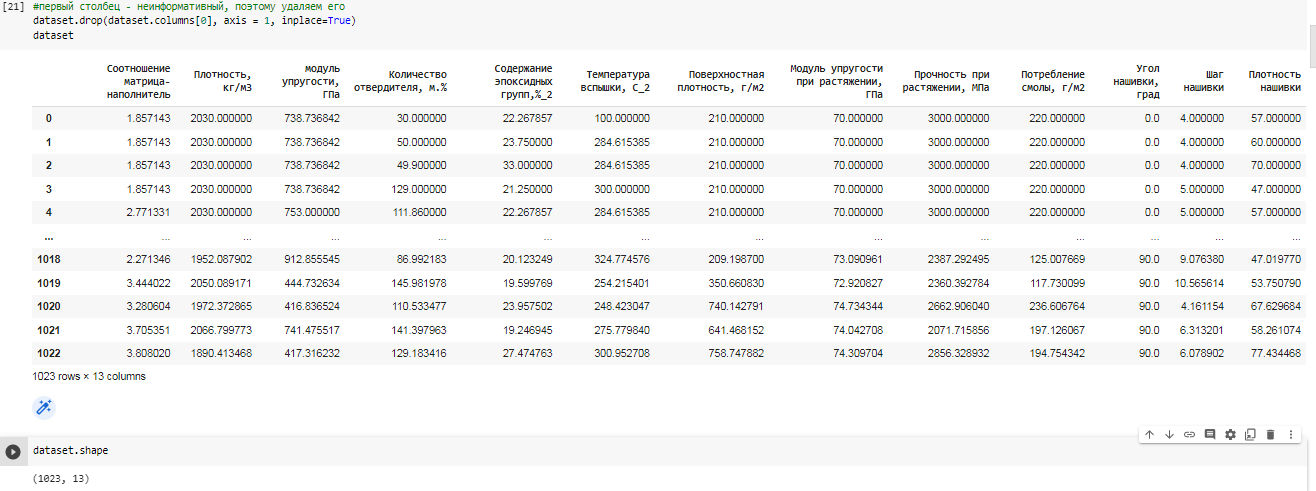


Рисунок 3 – Объединенный датасет без первого столбца

В качестве инструментов разведочного анализа используется:

1. общая информация о датасете:

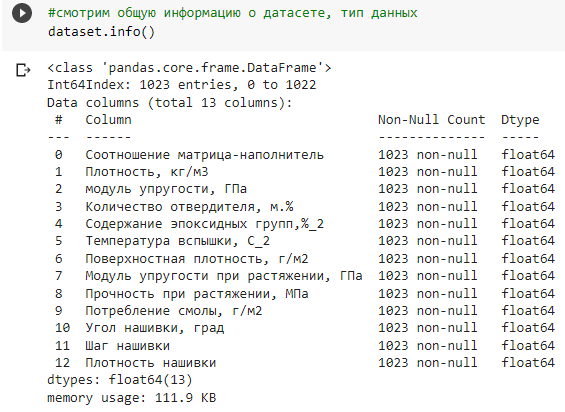


Рисунок 4 – Общая информация о датасете

Данный датасет содержит переменные значения float64, качественных характеристик нет, пропуски отсутствуют, ни одна из записей не является NaN, очистка не требуется.

1. описательная статистика для каждой переменной:

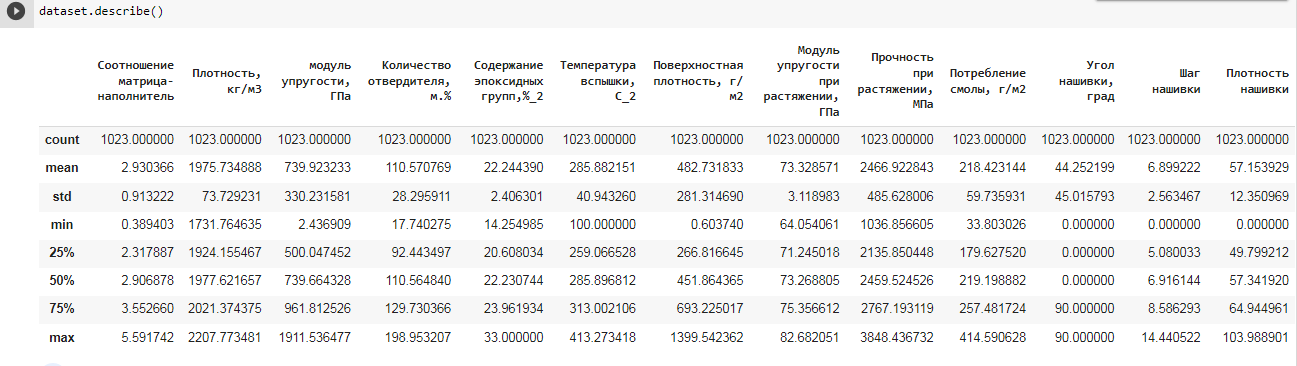


Рисунок 5 – Описательная статистика датасета

Описательная статистика содержит по каждому столбцу:

* count – количество значений
* mean – среднее значение
* std – стандартное отклонение
* min – минимум
* 25% – верхнее значение первого квартиля
* 50% – медиана
* 75% – верхнее значение третьего квартиля
* max – максимум

1. проверка наличия пропусков и дубликатов:

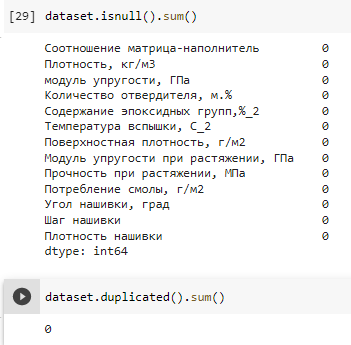


Рисунок 6 – Пропуски и дубликаты

В данном датасете отсутствуют пропуски и дубликаты, нулевых значений нет, следовательно, очистка не требуется.

При проверке на уникальность значений получаем, что столбец «Угол нашивки, град» имеет только два значения.

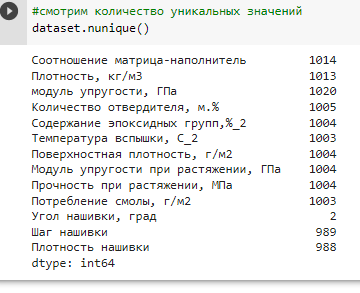


Рисунок 7 – Проверка уникальности значений

1. среднее и медианное значение:

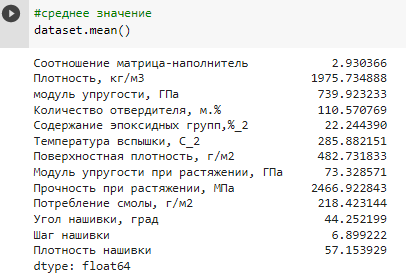


Рисунок 8 – Средние значения столбцов

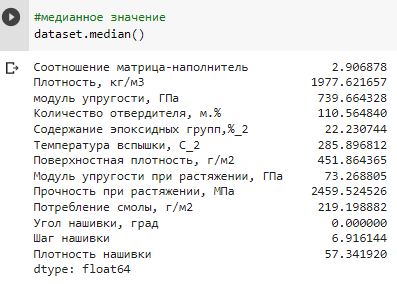


Рисунок 9 – Медианное значение столбцов

При анализе полученных результатов видим значения, близкие друг к другу. Только столбец «Угол нашивки, град» имеет сильно отличающиеся значения.

1. гистограммы распределения каждой из переменной:

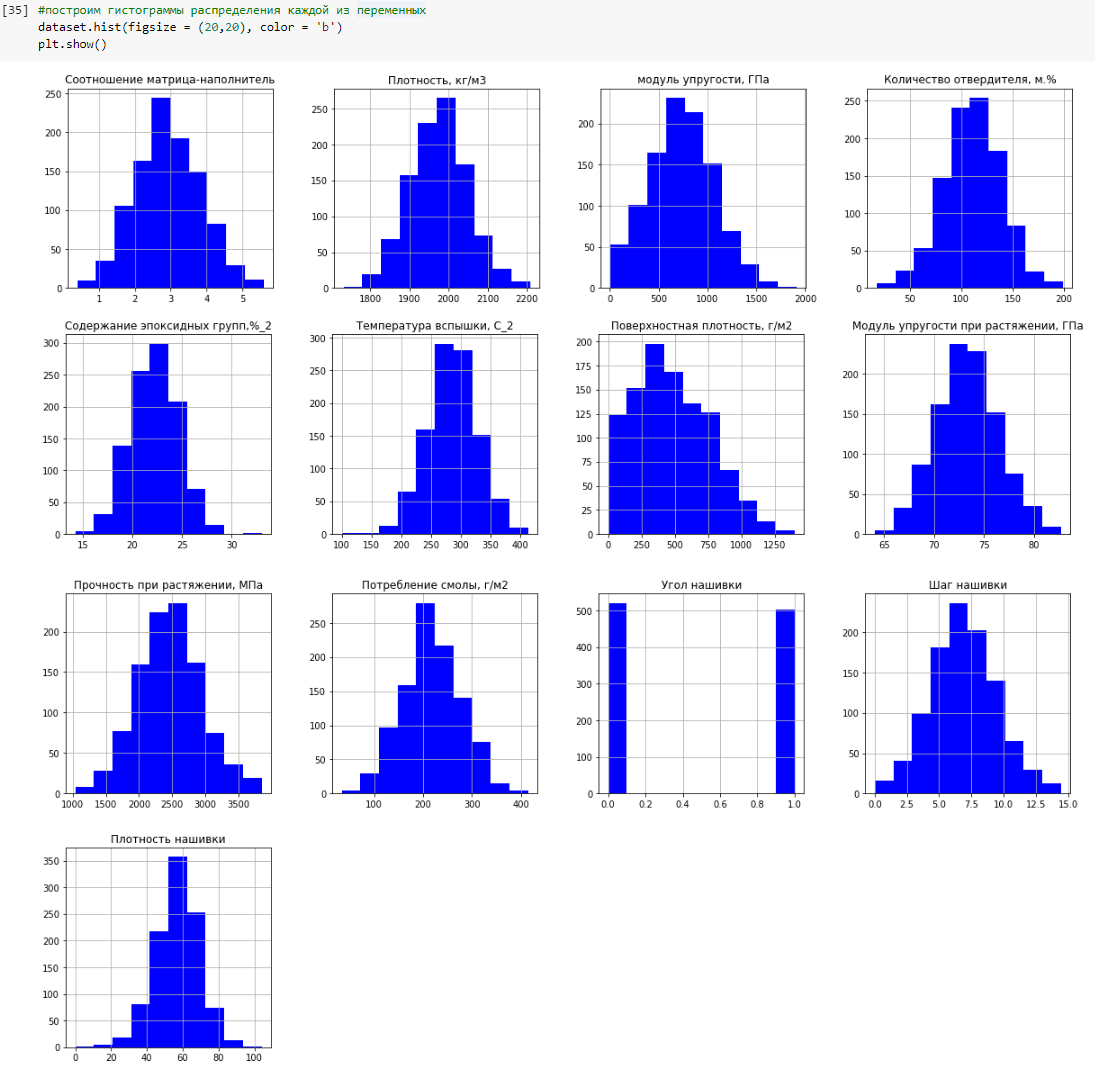


Рисунок 10 – Гистограммы распределения каждой переменной

При проведении анализа получаем, что данные преимущественно стремятся к нормальному распределению. Угол нашивки, как и отражено в датасете, имеет только два значения. Учитывая отсутствие иных показателей для угла нашивки, в прогнозе будем использовать не непрерывный, а категориальный подход при анализе данного параметра.

1. диаграммы «ящика с усами», анализ выбросов:

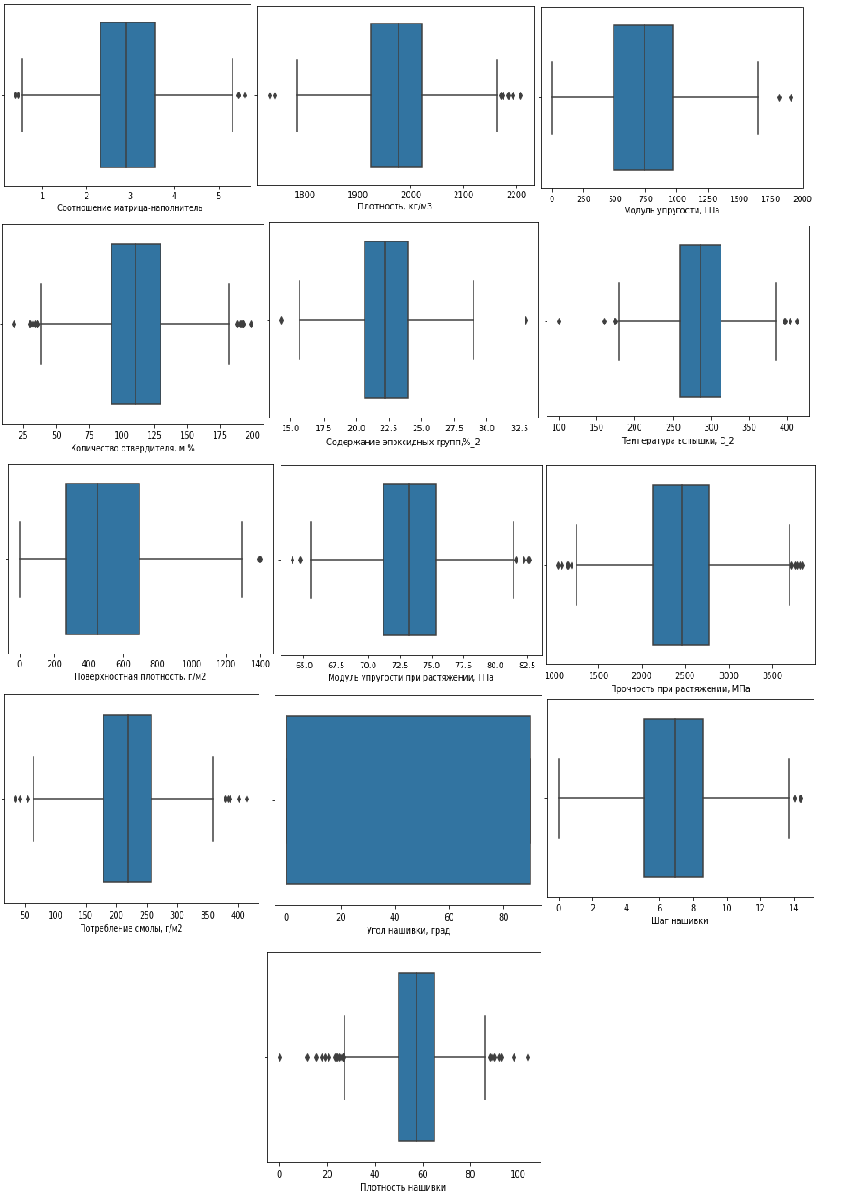


Рисунок 11 – «Ящики с усами»

Данные диаграммы показывают наличие выбросов данных во всех столбцах кроме столбца «Угол нашивки». Следовательно, в дальнейшем необходимо поработать с данными столбцов и исключить выбросы.

1. попарные графики рассеяния точек относительно целевых переменных:



Рисунок 12 – Попарные графики рассеяния точек относительно переменной «Модуль упругости при растяжении»

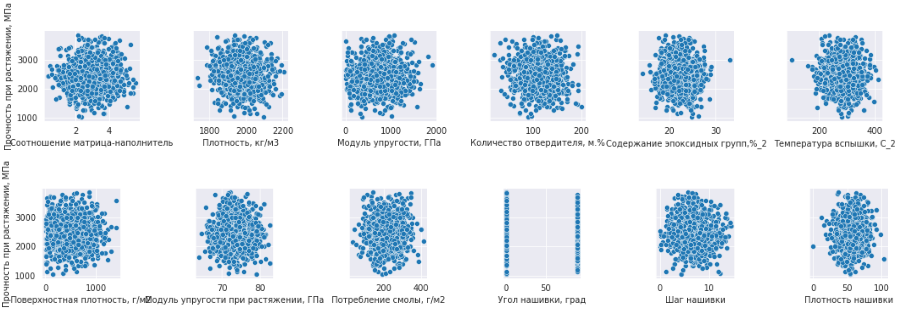


Рисунок 13 – Попарные графики рассеяния точек относительно переменной «Прочность при растяжении»

Данные графики не показывают какой-либо линейной зависимости между данными. Кроме того, можно наблюдать выбросы, т.к. некоторые точки располагаются далеко от общего облака. Можно сделать предположение об отсутствии линейной корреляции.

1. тепловая карта корреляционной зависимости:

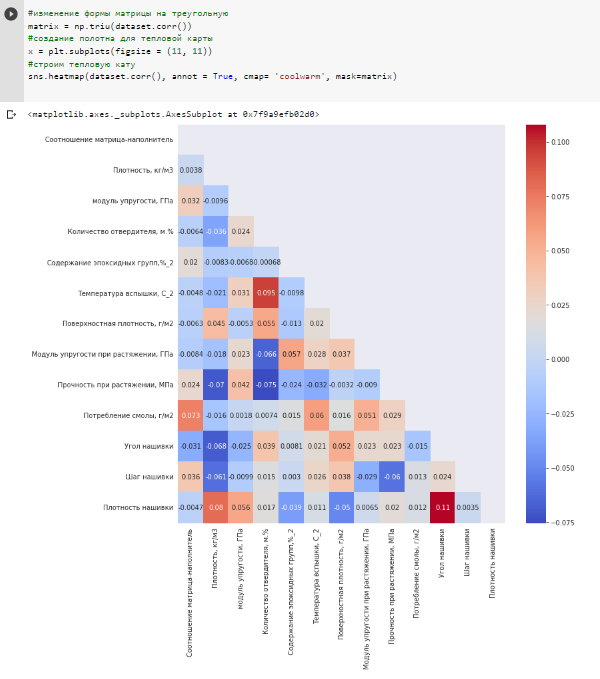


Рисунок 14 – Тепловая карта корреляционной зависимости

Проводя анализ тепловой карты, видим максимальное значение корреляции между «Плотностью нашивки» и «Углом нашивки» равно 0,11. Все остальные значения близки к 0. Можем сделать вывод об отсутствии корреляционной зависимости между данными.

Мы получили первоначальных представлений о характерах распределений переменных исходного набора данных, формирование оценки качества исходных данных: отсутствие пропусков, дубликатов, наличие выбросов в каждом столбце кроме столбца «Угол нашивки, град»; сделали вывод об отсутствии явной корреляционной зависимости между данными. На данном этапе мы предполагаем, что зависимость между данными все-таки есть. Выбираем подходящие методы решения – построение и обучение модели методом «Случайный лес», «К-ближайших соседей», «Дерево решений».

## **Практическая часть**

### **Предобработка данных**

На основе проведенного разведочного анализа меняем значения столбца «Угол нашивки, град»: 0° – 0, 90° – 1 и переименовываем его.

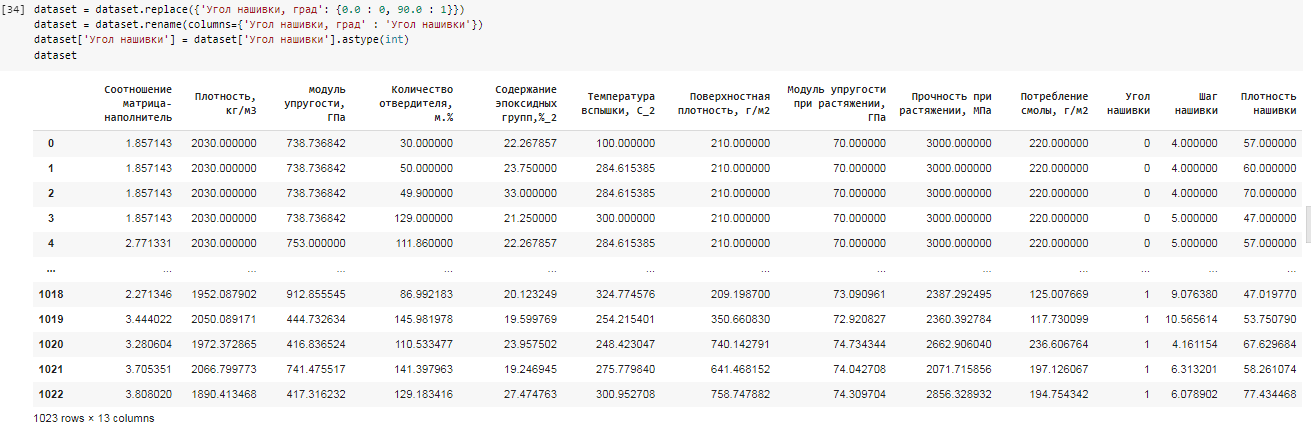


Рисунок 15 – Измененный датасет

Вновь смотрим среднее и медианное значения переменных. Убеждаемся в их незначительном отличии.

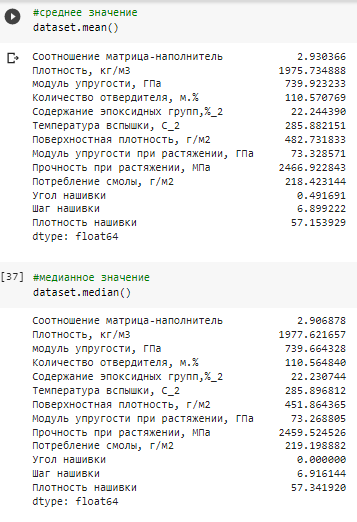


Рисунок 15 – Среднее и медианное значения

На этапе разведочного анализа, мы убедились, что в датасете присутствуют выбросы. Посчитаем их количество методом межквартильных расстояний (значения, удаленные более чем на 1,5 межквартильных расстояния, обнулим:

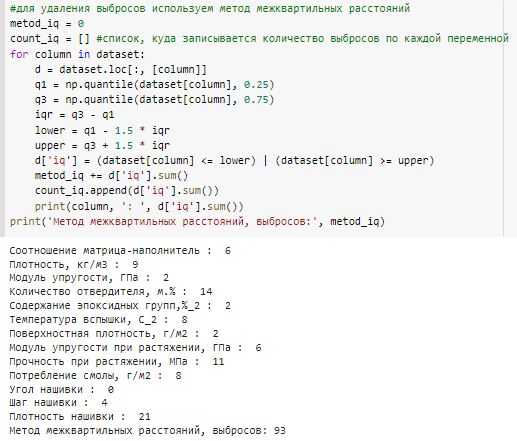


Рисунок 16 – Количество выбросов

Для удаления выбросов создаем переменную с названиями столбцов за исключением целевых переменных. Далее проводим удаление выбросов. Достигли результата за 3 подхода.

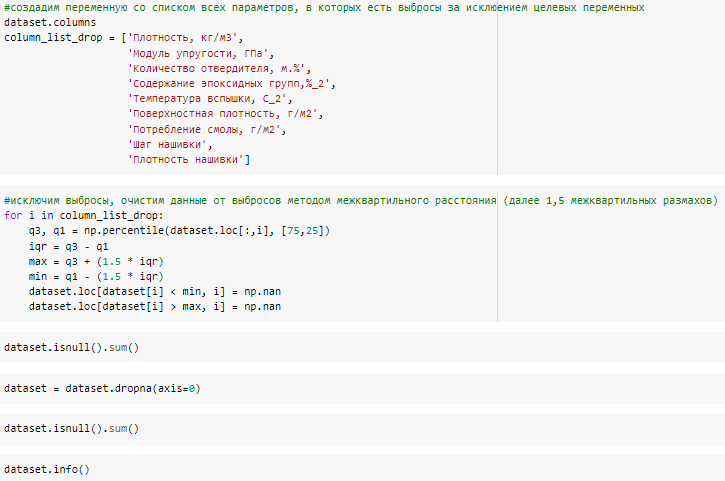


Рисунок 17 – Удаление выбросов

Визуализируем наши данные. Убеждаемся в их разбросе и в отсутствии корреляции. Следовательно, нужно данные отмасштабировать.

Масштабирование проводим следующими методами: MinMaxScaler, Normalizer, StandardScaler.

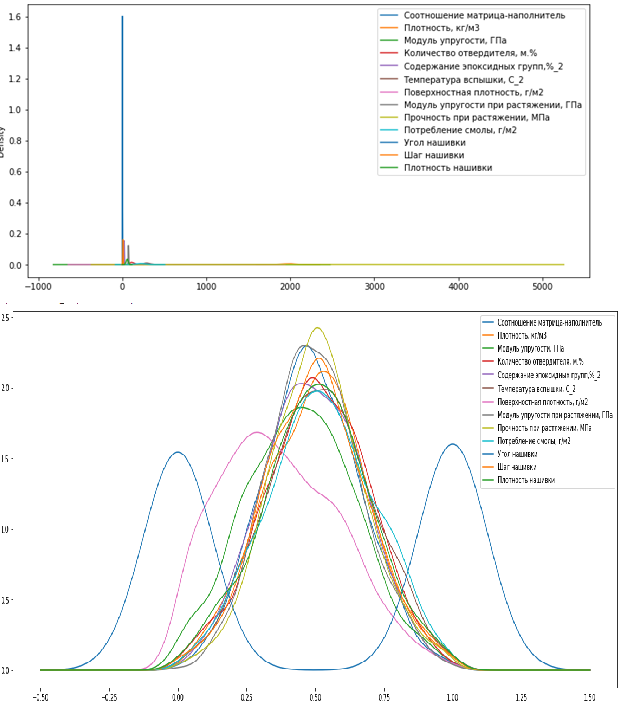


Рисунок 18 – Распределение данных

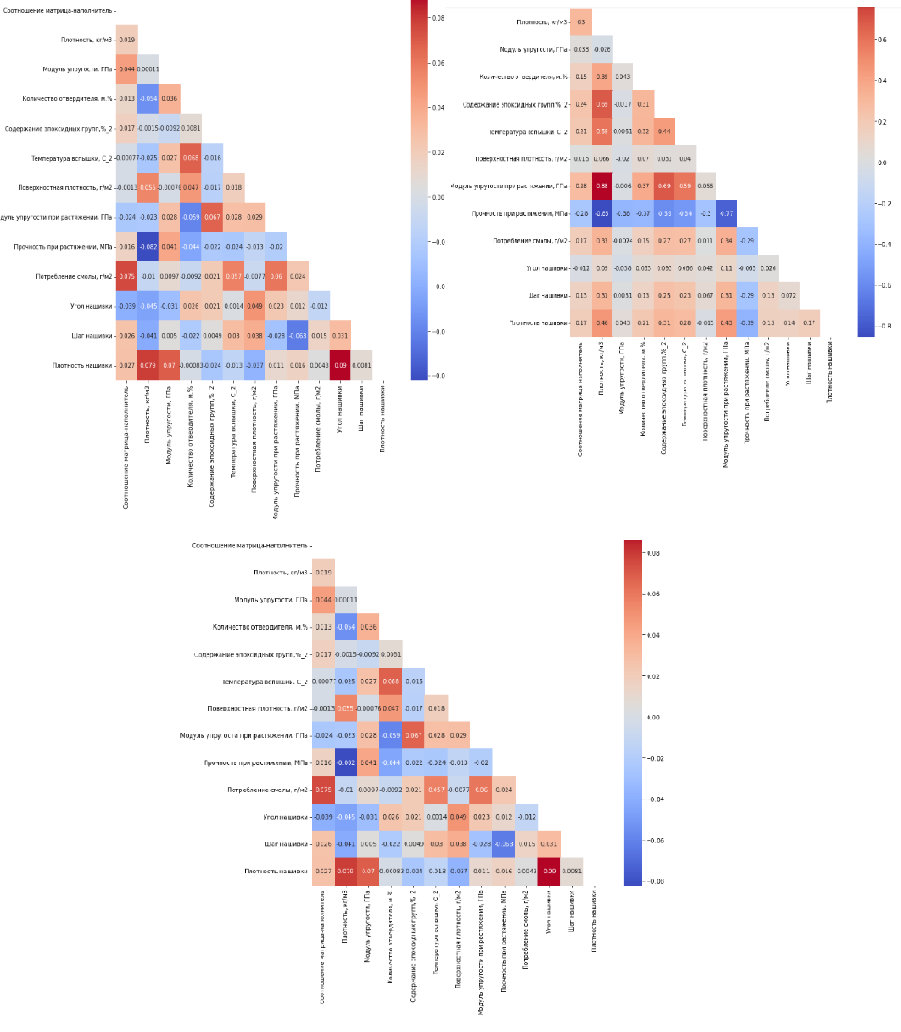


Рисунок 19 – Тепловые карты отмасштабированных датасетов

Наибольшая зависимость между данными в датасете после Normalizer. Поэтому именно с ним проводим дальнейшую работу.

### **Разработка и обучение модели**

Для разработки моделей выбрано 3 метода: RandomForest, K-Nearest Neighbors, Decision Trees. Так как у нас 2 целевые переменные, они не должны участвовать в разработке и обучении моделей друг друга.

Отдельно создаем модели для переменной «Модуль упругости при растяжении» и «Прочность при растяжении».

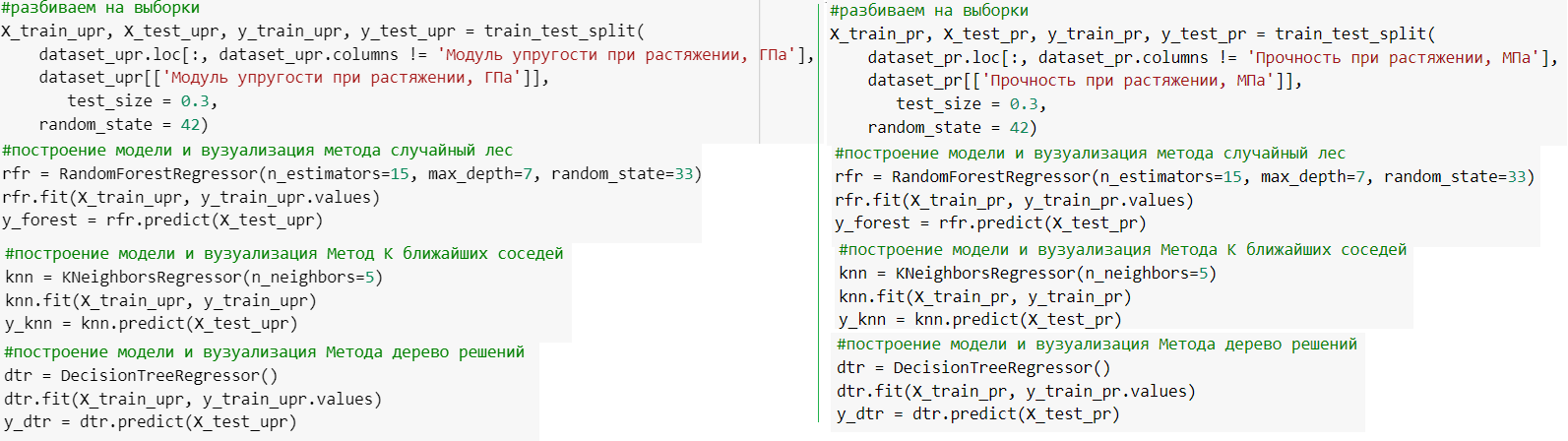


Рисунок 20 – Создание моделей

### **Тестирование модели**

Тестируем модели, визуализируем результат.

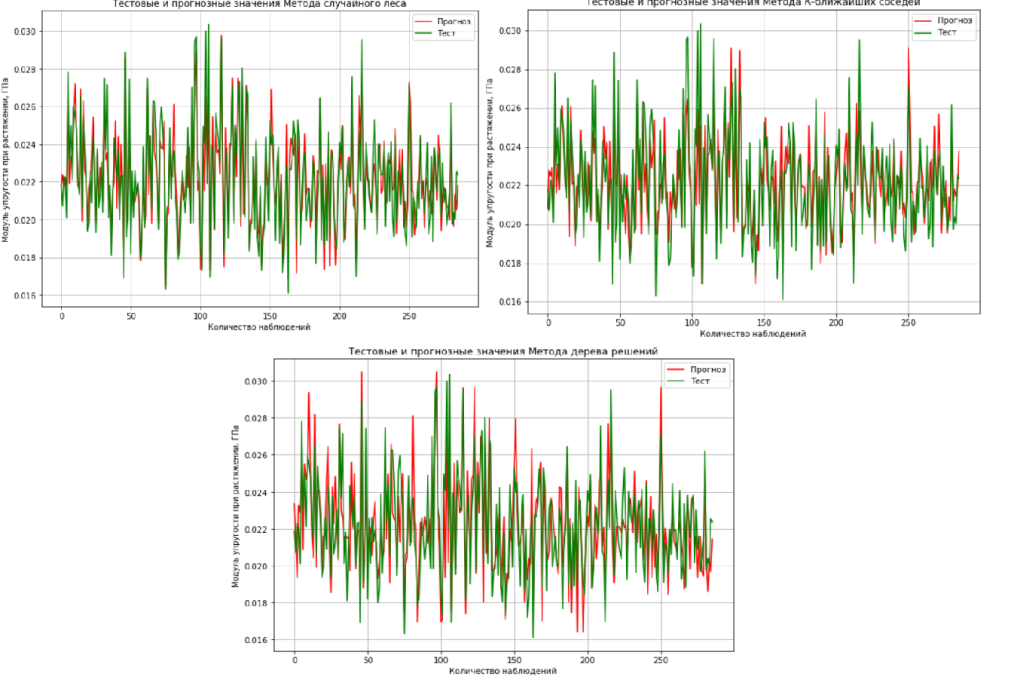


Рисунок 21 – Тестовые и прогнозные значения переменной «Модуль упругости при растяжении» для разных моделей

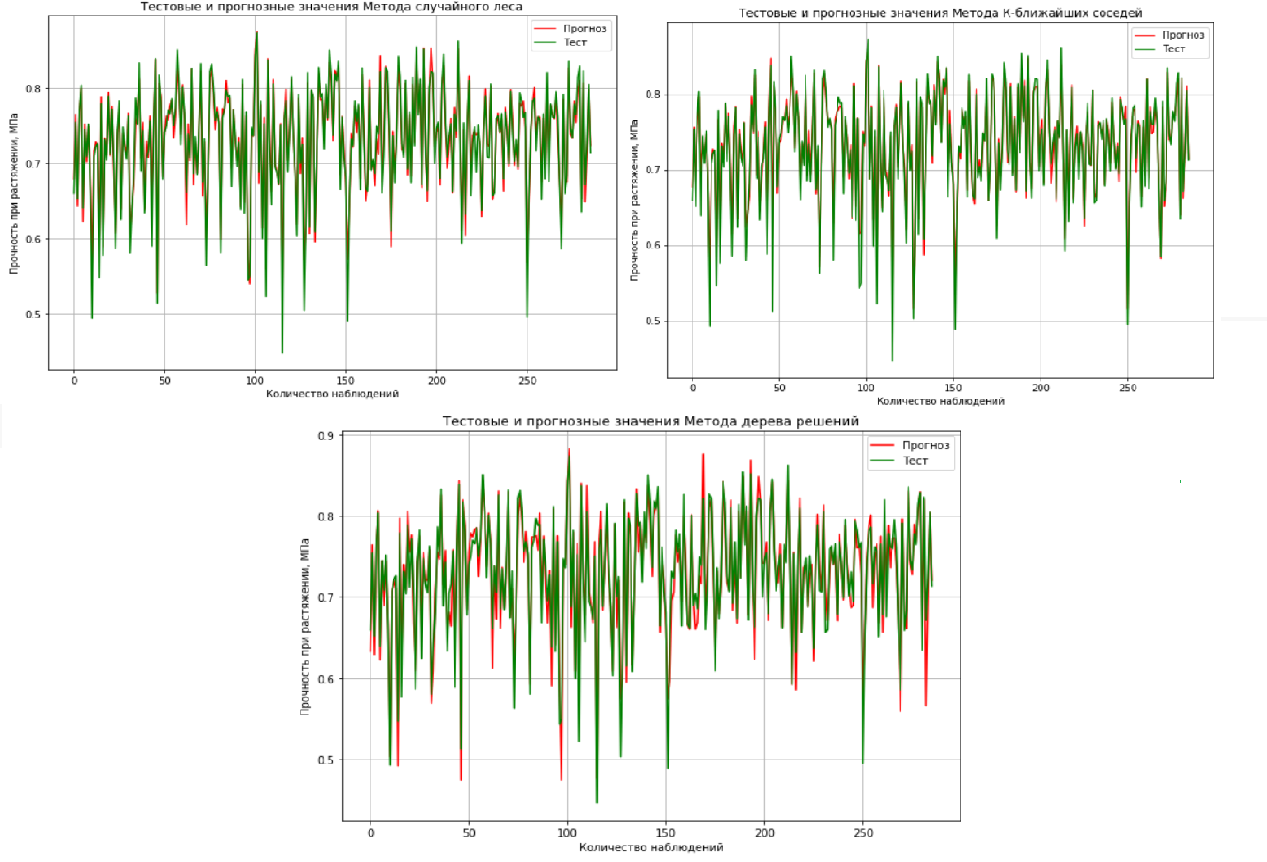


Рисунок 22 – Тестовые и прогнозные значения переменной «Прочность при растяжении» для разных моделей

Также с помощью метода GridSearchCV подбираем гиперпараметры, тестируем модели.



Рисунок 23 – Подбор гиперпараметров

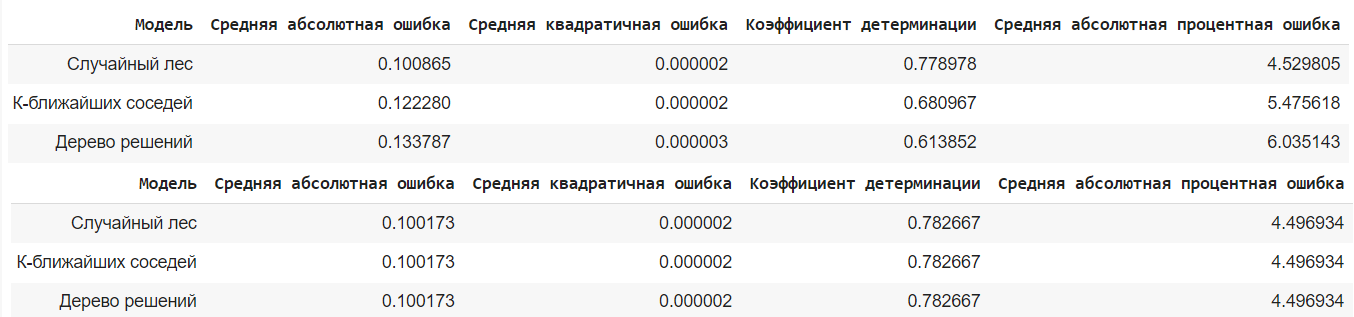


Рисунок 24 – Ошибки моделей для переменной «Модуль упругости при растяжении» до и после подбора гиперпараметров

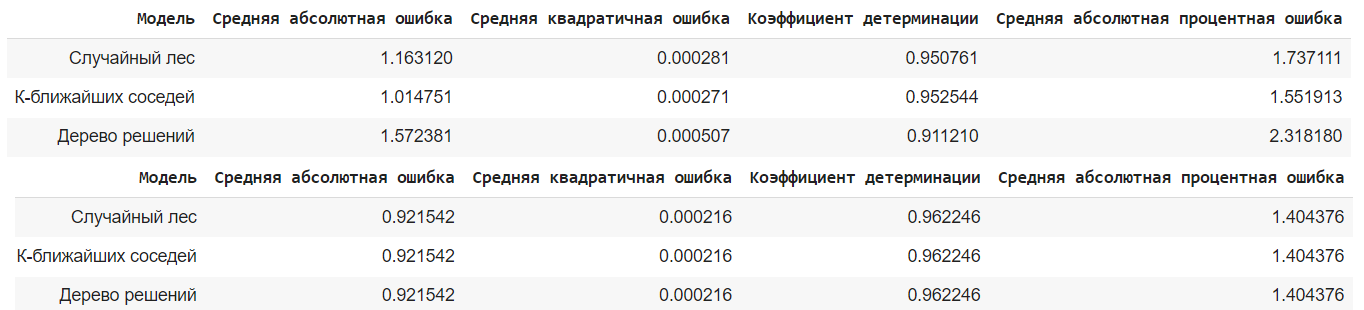


Рисунок 25 – Ошибки моделей для переменной «Прочность при растяжении» до и после подбора гиперпараметров

Средняя квадратичная ошибка стремится к нулю, коэффициент детерминации – к единице. Следовательно, можно говорить наших моделей.

### **Создание нейронной сети**

Создаем нейронную сеть с помощью Sequential.

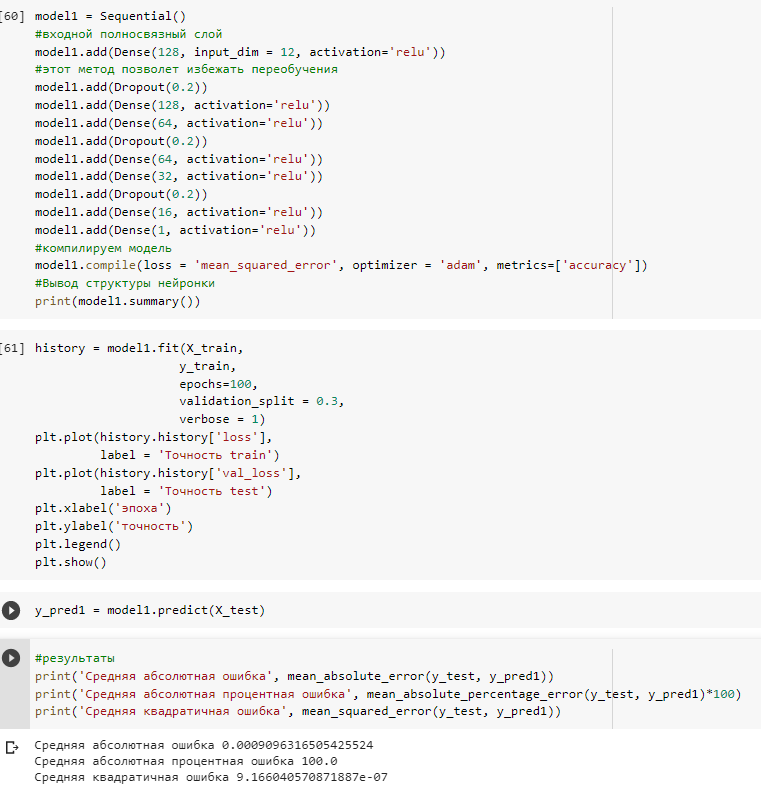


Рисунок 26 – Создание нейронной сети

По значениям ошибок можно сказать, что созданная нейронная сеть неуспешна.

### **Разработка приложения**



Рисунок 27 – Разработка приложения

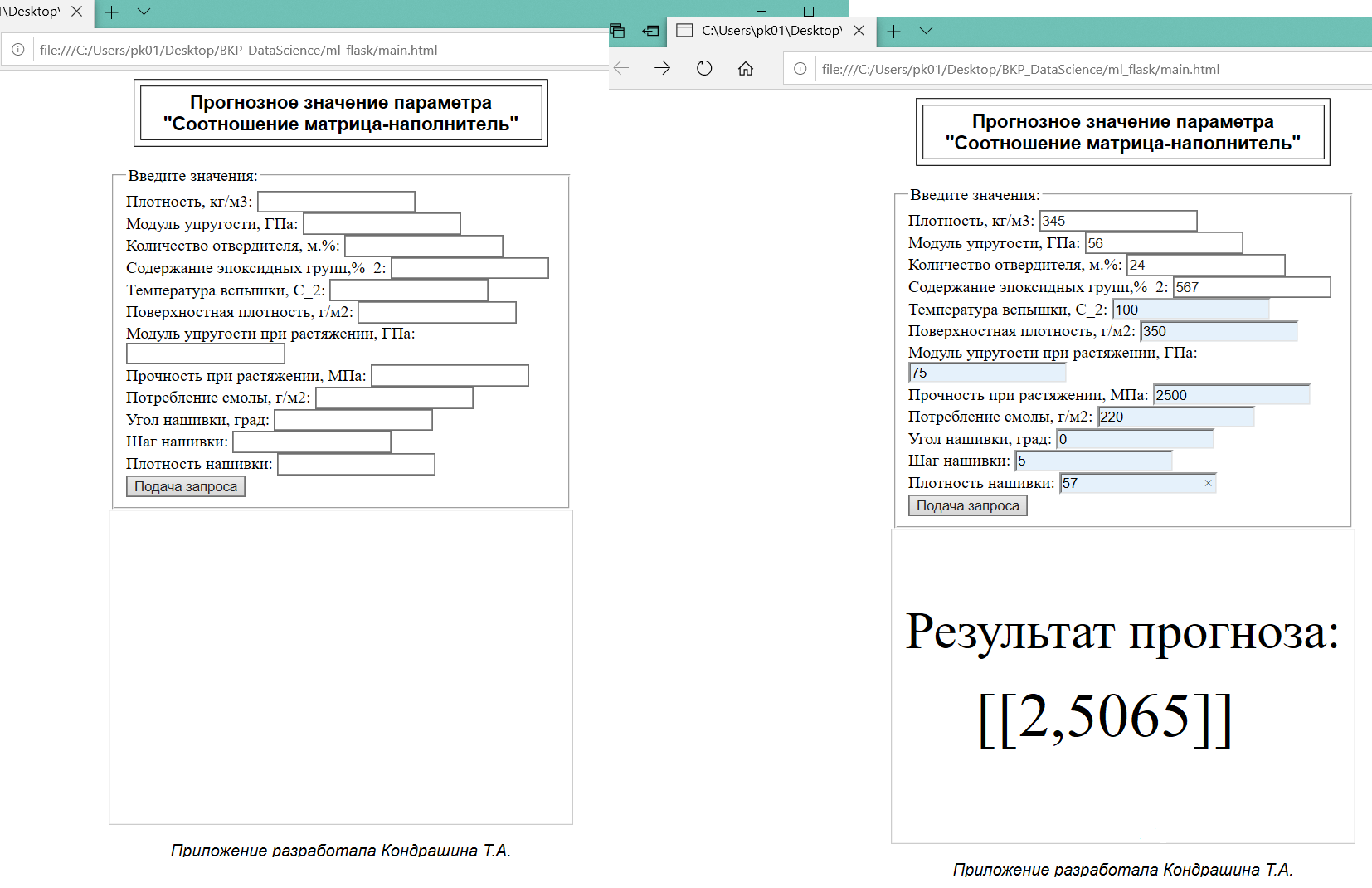


Рисунок 28 – Приложение

### **Создание удаленного репозитория и загрузка результатов работы на него**

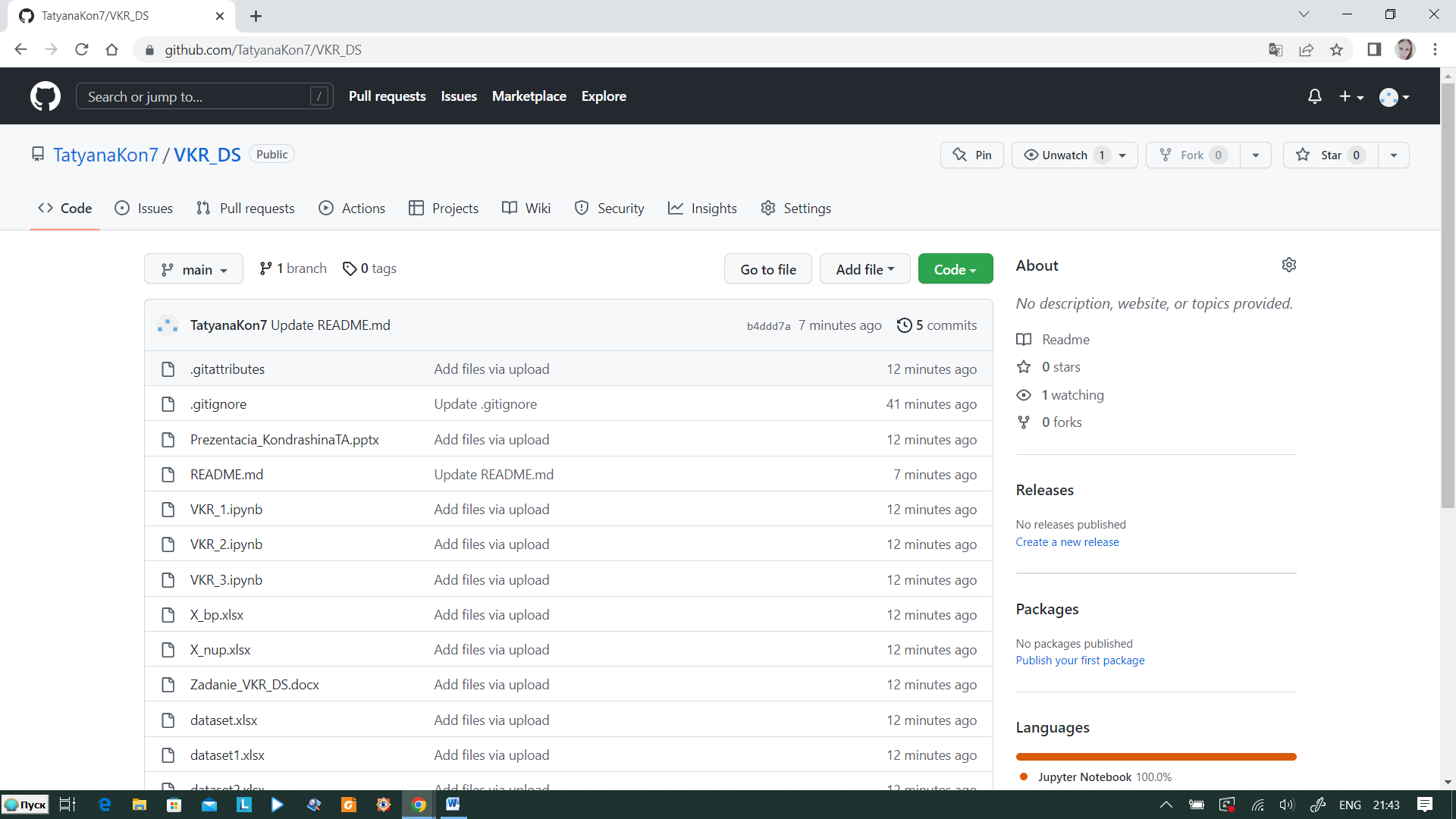


Рисунок 28 – Удаленный репозиторий

# 

# **Заключение**

В ходе исследовательской работы получены следующие результаты:

* распределение данных в объединённом датасете близко к нормальному
* коэффициенты корреляции между парами признаков стремятся к нулю, явной зависимости нет
* разработанные модели не обладают эффективностью

Предположение о наличии зависимости оказалось ложным. Но утверждать о том, что прогнозирование на данном датасете невозможно, не стоит. Можно предположить, что гипотеза не подтвердилась, по нескольким причинам:

* недостаточное количество данных, параметров;
* неудачные подходы при прогнозировании, моделировании;
* недостаток в знании теории копмозиционных материалов.

# **Список использованной литературы**

1. Бизли Д. Python. Подробный справочник: учебное пособие. – Пер. с англ. – СПб.: Символ-Плюс, 2010. – 864 с., ил.
2. 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.
3. Грас Д. Data Science. Наука о данных с нуля: Пер. с англ. - 2-е изд., перераб. и доп. - СПб.: БХВ-Петербурr, 2021. - 416 с.: ил.
4. Документация по библиотеке keras: – Режим доступа: <https://keras.io/api/>. (дата обращения: 08.11.2022).
5. Документация по библиотеке matplotlib: – Режим доступа: <https://matplotlib.org/stable/users/index.html>. (дата обращения: 07.11.2022).
6. Документация по библиотеке numpy: – Режим доступа: <https://numpy.org/doc/1.22/user/index.html#user>. (дата обращения: 03.11.2022).
7. Документация по библиотеке pandas: – Режим доступа: <https://pandas.pydata.org/docs/user_guide/index.html#user-guide>. (дата обращения: 04.11.2022).
8. Документация по библиотеке scikit-learn: – Режим доступа: <https://scikit-learn.org/stable/user_guide.html>. (дата обращения: 07.11.2022).
9. Документация по библиотеке seaborn: – Режим доступа: <https://seaborn.pydata.org/tutorial.html>. (дата обращения: 07.11.2022).
10. Документация по библиотеке Tensorflow: – Режим доступа: <https://www.tensorflow.org/overview> (дата обращения: 07.11.2022).
11. Жерон, Орельен. Прикладное машинное обучение с помощью Scikit-Learn и TensorFlow: концепции, инструменты и техники для создания интеллектуальных систем. Пер. с англ. - СпБ.: ООО "Альфа-книга': 2018. - 688 с.: ил.
12. Руководство по быстрому старту в flask: – Режим доступа: <https://flask-russian-docs.readthedocs.io/ru/latest/quickstart.html>. (дата обращения: 08.11.2022).

# **Приложение 1**

**План исследования**

1. Загружаем и обрабатываем входящие датасеты.
   1. Удаляем неинформативные столбцы.
   2. Объединяем датасеты по методу INNER.
2. Проводим разведочный анализ данных:
   1. Изучим описательную статистику каждой переменной – среднее, медиана, стандартное отклонение, минимум, максимум, квартили.
   2. Проверим датасет на пропуски и дубликаты данных
3. Визуализируем наш разведочный анализ данных (до выбросов и нормализации).
   1. Построим гистограмм распределения переменных.
   2. Построим диаграммы ящиков с усами каждой переменной.
   3. Построим попарные графики рассеяния точек относительно целевых переменных.
   4. Построим корреляционную матрицу с помощью тепловой карты.
   5. Сохраняем датасет.
4. Проведём предобработку данных.
   1. Меняем название столбца «Угол нашивки, град» на «Угол нашивки» и его значения: 0 на 0, 90 – 1.
   2. Проверим выбросы методом межквартильных расстояний.
   3. Посчитаем распределение выбросов по каждому столбцу.
   4. Создаем переменную с названиями столбцов за исключением целевых переменных.
   5. Исключим выбросы методом межквартильных расстояний (обнулим значения выбросов).
   6. Удалим строки c выбросами.
   7. Для полной очистки датасета от выбросов повторим пункты (4.5 – 4.6) ещё 3 раза.
   8. Изучим чистые данные по всем параметрам.
   9. Визуализируем датасет без выбросов.
5. Проведём нормализацию и стандартизацию (продолжим предобработку данных).
   1. Нормализуем данные с помощью MinMaxScaler().
   2. Нормализуем данные с помощью Normalizer().
   3. Визуализируем данные.
   4. Сравним с данными до нормализации.
   5. Строим тепловую карту корреляции.
   6. Стандартизируем данные
   7. Визуализируем данные корреляции
   8. Посмотрим на описательную статистику после нормализации и после стандартизации
6. Разработаем и обучим нескольких моделей прогноза модуля упругости при растяжении (с 30% тестовой выборки)
   1. Определим входы и выходы для моделей.
   2. Разобьём данные на обучающую и тестовую выборки.
   3. Проверим правильность разбивки.
   4. Построим и визуализируем результат работы метода случайного леса.
   5. Построим и визуализируем результат работы метода К ближайших соседей.
   6. Построим и визуализируем результат работы метода деревья решений.
   7. Сравним наши модели по метрикам.
   8. Найдём лучшие гиперпараметры для случайного леса.
   9. Подставим значения в нашу модель случайного леса.
   10. Найдём лучшие гиперпараметры для К ближайших соседей.
   11. Подставим значения в нашу модель К ближайших соседей.
   12. Найдём лучшие гиперпараметры метода деревья решений.
   13. Подставим значения в нашу модель метода деревья решений.
   14. Сравним все модели и процессинги и выведем лучшую модель и процессинг.
7. Разработаем и обучим нескольких моделей прогноза прочности при растяжении (с 30% тестовой выборки).
   1. Определим входы и выходы для моделей.
   2. Разобьём данные на обучающую и тестовую выборки.
   3. Проверим правильность разбивки.
   4. Построим и визуализируем результат работы метода случайного леса.
   5. Построим и визуализируем результат работы метода К ближайших соседей.
   6. Построим и визуализируем результат работы метода деревья решений.
   7. Сравним наши модели по метрикам.
   8. Найдём лучшие гиперпараметры для случайного леса.
   9. Подставим значения в нашу модель случайного леса.
   10. Найдём лучшие гиперпараметры для К ближайших соседей.
   11. Подставим значения в нашу модель К ближайших соседей.
   12. Найдём лучшие гиперпараметры метода деревья решений.
   13. Подставим значения в нашу модель метода деревья решений.
   14. Сравним все модели и процессинги и выведем лучшую модель и процессинг.
8. Нейронная сеть для рекомендации соотношения матрица-наполнитель.
   1. Сформируем входы и выход для модели.
   2. Нормализуем данные.
   3. Построим модель, определим параметры.
   4. Посмотрим на результаты.
   5. Обучим нейросеть 70/30.
   6. Оценим модель.
   7. Посмотрим на график потерь на тренировочной и тестовой выборках.
   8. Сконфигурируем другую модель, зададим слои.
   9. Посмотрим на архитектуру другой модели.
   10. Обучим другую модель.
   11. Оценим модель.
   12. Посмотрим на график потерь на тренировочной и тестовой выборках.
   13. Сохраняем вторую модель для разработки веб-приложения для прогнозирования соотношения «матрица-наполнитель» в фреймворке Flask.
9. Создаём приложение.
   1. Импортируем необходимые бибилиотеки.
   2. Загрузим модель и определим параметры функции
   3. Получим данные из наших форм и положим их в список.
   4. Укажем шаблон и прототип сайта для вывода.
   5. Запустим приложение.
10. Создание удалённого репозитория и загрузка результатов работы на него.
    1. <https://github.com/TatyanaKon7/VKR_DS>
    2. Создадим README (<https://github.com/TatyanaKon7/VKR_DS/blob/main/README.md>)
    3. Выгрузим все необходимые файлы в репозиторий.