МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

**по курсу**

«Data Science»

**Тема:** «**Прогнозирование конечных свойств новых материалов**

**(композиционных материалов)»**

Слушатель Малакаев Шапи Жабраилович

Москва, 2023

**Содержание**

[Введение 3](#_TOC_250034)

1. [Аналитическая часть 4](#_TOC_250033)
   1. [Постановка задачи 4](#_TOC_250032)
   2. [Описание используемых методов 5](#_TOC_250031)
   3. [Разведочный анализ данных 14](#_TOC_250022)
2. [Практическая часть 24](#_TOC_250015)
   1. [Предобработка данных 24](#_TOC_250014)
   2. [Разработка и обучение моделей 28](#_TOC_250010)
   3. [Тестирование моделей 32](#_TOC_250005)
   4. [Нейронная сеть 33](#_TOC_250004)
   5. Разработка приложения 37
   6. [Создание удаленного репозитория 37](#_TOC_250003)

[Заключение 38](#_TOC_250002)

[Библиографический список 39](#_TOC_250001)

# 

# Введение

Тема данной работы - прогнозирование конечных свойств новых материалов (композиционных материалов).

Композиционными называются материалы, в которых имеет место сочетание двух (или более) химически разнородных компонентов (фаз) с четкой границей раздела между ними. Это неоднородные по химическому составу и структуре материалы.

Структура композиционных материалов представляет собой матрицу (основной компонент), содержащую в своем объеме или армирующие элементы, часто называемые наполнителем. Матрица и наполнитель разделены границей (поверхностью) раздела. Наполнитель равномерно распределен в матрице и имеет заданную пространственную ориентацию.

Композиционные материалы характеризуются совокупностью свойств, не присущих каждому в отдельности взятому компоненту. За счет выбора армирующих элементов, варьирования их объемной доли в матричном материале, а также размеров, формы, ориентации и прочности связи по границе «матрица- наполнитель», свойства композиционных материалов можно регулировать в значительных пределах.

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

Учитывая широкое распространение и высокую потребность в новых материалах, способных дать конкурентные преимущества, тема данной работы является актуальной.

Стоимость производства композитного материала высока. Для получения заданных свойств требуется большое количество дорогих испытаний. Сократить затраты на создание композитов прогнозированием их конечных свойств при помощи методов машинного обучения - актуальная задача.

# Аналитическая часть

* 1. **Постановка задачи**

В данной работе исследуется композит с матрицей из базальтопластика и нашивками из углепластика. Дан датасет, содержащий данные о свойствах матрицы и наполнителя, производственных параметрах и свойствах готового композита. Требуется разработать модели, прогнозирующие значения некоторых свойств в зависимости от остальных. Так же требуется разработать приложение, делающее удобным использование данных моделей специалистами в предметной области.

Датасет состоит из двух файлов: X\_bp (составляющая из базальтопластика) и Х\_nup (составляющая из углепластика).

Файл X\_bp содержит:

* признаков: 10;

- строк: 1023.

Файл X\_nup содержит:

* признаков: 3;

- строк: 1040.

Файлы требуется объединить по типу INNER. После объединения часть строк из файла X\_nup отброшена. И дальнейшие исследования проводим с объединенным датасетом, содержащим 13 признаков и 1023 строк.

Также необходимо провести разведочный анализ данных, нарисовать гистограммы распределения каждой из переменной, диаграммы boxplot (ящик с усами), попарные графики рассеяния точек.

Для каждой колонки получить среднее, медианное значение, провести анализ и исключение выбросов, проверить наличие пропусков; сделать предобработку: удалить шумы и выбросы, сделать нормализацию и стандартизацию.

Обучить несколько моделей для прогноза модуля упругости при растяжении и прочности при растяжении. Написать нейронную сеть, которая будет рекомендовать соотношение матрица-наполнитель. Разработать приложение с графическим интерфейсом, которое будет выдавать прогноз соотношения «матрица-наполнитель». Оценить точность модели на тренировочном и тестовом датасете. Создать репозиторий в GitHub и разместить код исследования. Оформить файл README

**1.2 Описание используемых методов**

Прогнозирование значений вещественной, непрерывной величины — это задача регрессии. прогнозируемая величина должна иметь связь с одной или несколькими независимыми переменными, называемых также предикторами или регрессорами. Регрессионный анализ помогает понять, как «типичное» значение прогнозируемой величины изменяется при изменении независимых переменных.

В настоящее время разработано много методов регрессионного анализа. Рассмотрим следующие.

.

**1.2.1Линейная регрессия**

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

*y = a\*x + b* (1)

Коэффициенты a и b, называемые также параметрами модели, определяются таким образом, чтобы сумма квадратов отклонений точек, соответствующих реальным наблюдениям данных, от линии регрессии была бы минимальной. Коэффициенты обычно оцениваются методом наименьших квадратов.

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

*y = b0 + b1 \* x1 + b2 \* x2 +…+ bn \* xn,* (2)

где n - число входных переменных.

Очевидно, что в данном случае модель будет описываться не прямой, а гиперплоскостью. Коэффициенты уравнения множественной линейной регрессии подбираются так, чтобы минимизировать сумму квадратов отклонения реальных точек данных от этой гиперплоскости.

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

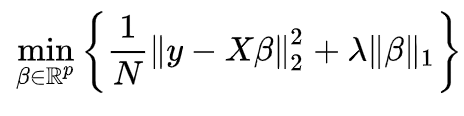
* + 1. **Регрессия Лассо (LASSO)**

Это линейная модель, которая оценивает разреженные коэффициенты. Это простой метод, позволяющий уменьшить сложность модели и предотвратить переопределение, которое может возникнуть в результате простой линейной регрессии. Данный метод вводит дополнительное слагаемое регуляризации в оптимизацию модели. Это даёт более устойчивое решение. В регрессии лассо добавляется условие смещения в функцию оптимизации для того, чтобы уменьшить коллинеарность и, следовательно, дисперсию модели. Но вместо квадратичного смещения, используется смещение абсолютного значения. Лассо регрессия хорошо прогнозирует модели временных рядов на основе регрессии, таким как авторегрессии.

*Достоинства метода:* легко полностью избавляется от шумов в данных; быстро работает; не очень энергоёмко; способно полностью убрать признак из датасета; доступно обнуляет значения коэффициентов.

*Недостатки метода:* часто страдает качество прогнозирования; выдаёт ложное срабатывание результата; случайным образом выбирает одну из коллинеарных переменных; не оценивает правильность формы взаимосвязи между независимой и зависимой переменными; не всегда лучше, чем пошаговая регрессия.

Лассо-регрессию следует использовать, когда есть несколько характеристик с высокой предсказательной способностью, а остальные бесполезны. Она обнуляет бесполезные характеристики и оставляет только подмножество переменных. Регрессия Лассо имеет вид:

 (3)

* + 1. **Гребневая (Ridge) регрессия**

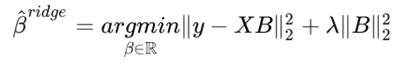
Это регрессия, которая добавляет дополнительный штраф к функции стоимости, но вместо этого суммирует квадраты значений коэффициентов (норма L-2) и умножает их на некоторую постоянную лямбду. По сравнению с Лассо этот штраф регуляризации уменьшит значения коэффициентов, но не сможет принудительно установить коэффициент равным 0. Это ограничивает использование регрессии гребня в отношении выбора признаков. Однако, когда p > n (p –количество предикторов, n – число наблюдений), он способен выбрать более n релевантных предикторов, если необходимо, в отличие от Лассо. Он также выберет группы коллинеарных элементов, которые его изобретатели назвали «эффектом группировки».

Как и в случае с Лассо, мы можем варьировать лямбду, чтобы получить модели с различными уровнями регуляризации, где лямбда = 0 соответствует OLS, а лямбда приближается к бесконечности, что соответствует постоянной функции.

Анализ регрессии Лассо, как и Ridge показывает, что ни одному из методов нельзя отдать предпочтение; нужно попробовать оба метода, чтобы определить, какой использовать.

Ridge-регрессию лучше применять, когда предсказательная способность набора данных распределена между различными характеристиками. Ridge-регрессия не обнуляет характеристики, которые могут быть полезны при составлении прогнозов, а просто уменьшает вес большинства переменных в модели.

Ridge-регрессия имеет вид:

 (4)

* + 1. **Эластичная сеть (Elastic Net)**

Это регрессия, которая включает в себя термины регуляризации как L-1, так и L-2. Это дает преимущества регрессии Лассо и Ridge. Было установлено, что он обладает предсказательной способностью лучше, чем у Лассо, хотя все еще выполняет выбор функций. Поэтому получается лучшее из обоих методов, выполняя выбор функции Лассо с выбором группы объектов Ridge.

Elastic Net поставляется с дополнительными издержками на определение двух лямбда-значений для оптимальных решений.

Компромисс смещения дисперсии - это компромисс между сложной и

простой моделью, в которой промежуточная сложность, вероятно, является наилучшей.

Elastic Net имеет вид:

 (5)

Лассо, Ridge-регрессия и Эластичная сеть - это модификации обычной линейной регрессии наименьших квадратов, которые используют дополнительные штрафные члены в функции стоимости, чтобы сохранить значения коэффициента небольшими и упростить модель.

Лассо полезно для выбора функций, когда наш набор данных имеет функции с плохой предсказательной силой.

Регрессия гребня полезна для группового эффекта, при котором коллинеарные элементы могут быть выбраны вместе.

Elastic Net сочетает в себе регрессию Лассо и Ridge, что потенциально приводит к модели, которая является простой и прогнозирующей.

* + 1. **Машина опорных векторов (Support Vector Machine)**

Метод опорных векторов (support vector machine, SVM) — один из наиболее популярных методов машинного обучения. Он создает гиперплоскость или набор гиперплоскостей в многомерном пространстве, которые могут быть использованы для решения задач классификации и регрессии.

Чаще всего он применяется в постановке бинарной классификации.

Основная идея заключается в построении гиперплоскости, разделяющей объекты выборки оптимальным способом. Хорошее разделение достигается за счет гиперплоскости, которая имеет самое большое расстояние до ближайшей точки обучающей выборке любого класса. Максимально близкие объекты разных классов определяют опорные вектора.

Если в исходном пространстве объекты линейно неразделимы, то выполняется переход в пространство большей размерности.

Решается задача оптимизации.

Для вычислений используется ядерная функция, получающая на вход два вектора и возвращающая меру сходства между ними:

* линейная;
* полиномиальная;
* гауссовская (rbf).

Эффективность метода опорных векторов зависит от выбора ядра, параметров ядра и параметра С для регуляризации.

Преимущество метода — его хорошая изученность. Недостатки:

* чувствительность к выбросам;
* отсутствие интерпретируемости.

Вариация метода для регрессии называется SVR (Support Vector Regression).

* + 1. **Метод k-ближайших соседей (K-Nearest Neighbors)**

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

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

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

Этот метод — пример непараметрической регрессии.

* + 1. **Решающее дерево (Decision Tree)**

Решающее дерево (Decision Tree) - еще один непараметрический метод, применяемый и для классификации, и для регрессии. Решающее дерево используются в самых разных областях человеческой деятельности и представляют собой иерархические древовидные структуры, состоящие из правил вида «Если ..., то ...».

Решающие правила автоматически генерируются в процессе обучения на обучающем множестве путем обобщения обучающих примеров. Поэтому их называют индуктивными правилами, а сам процесс обучения — индукцией деревьев решений.

Дерево состоит из элементов двух типов: узлов (node) и листьев (leaf).

В узлах находятся решающие правила и производится проверка соответствия примеров этому правилу. В результате проверки множество примеров, попавших в узел, разбивается на два подмножества: удовлетворяющие правилу и не удовлетворяющие ему. Затем к каждому подмножеству вновь применяется правило, и процедура рекурсивно повторяется пока не будет достигнуто некоторое условие остановки алгоритма. В последнем узле проверка и разбиение не производится, и он объявляется листом.

В листе содержится не правило, а подмножество объектов, удовлетворяющих всем правилам ветви, которая заканчивается данным листом. Для

классификации — это класс, ассоциируемый с узлом, а для регрессии — соответствующий листу интервал целевой переменной.

При формировании правила для разбиения в очередном узле дерева необходимо выбрать атрибут, по которому это будет сделано. Общее правило для классификации можно сформулировать так: выбранный атрибут должен разбить множество наблюдений в узле так, чтобы результирующие подмножества со-

держали примеры с одинаковыми метками класса, а количество объектов из других классов в каждом из этих множеств было как можно меньше. Для этого были выбраны различные критерии, например, теоретико-информационный и статистический.

Для регрессии критерием является дисперсия вокруг среднего. Минимизируя дисперсию вокруг среднего, мы ищем признаки, разбивающие выборку таким образом, что значения целевого признака в каждом листе примерно равны.

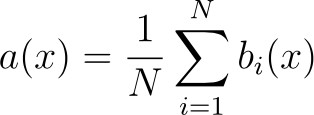
Огромное преимущество решающего дерева в том, что они легко интерпретируемы, понятны человеку. Они могут использоваться для извлечения правил на естественном языке. Еще преимущества — высокая точность работы, нетребовательность к подготовке данных.

Недостаток решающего дерева - большая склонность переобучаться. Переобучение в этом случае — это точное распознавание примеров, участвующих в обучении и полная несостоятельность на новых данных. В худшем случае, дерево будет большой глубины и сложной структуры, а в каждом листе будет только один объект. Для решения этой проблемы используют разные критерии остановки алгоритма.

* + 1. **Случайный лес (Random Forest)**

Случайный лес — представитель ансамблевых методов.

Если точность решающего дерева оказалось недостаточной, мы можем множество деревьев собрать в лес. Формула итогового решателя (6) — это усреднение предсказаний отдельных деревьев.

(6),

где

N – количество деревьев; i – счетчик для деревьев; b – решающее дерево;

x – сгенерированная нами на основе данных выборка.

Для определения входных данных каждому дереву используется метод случайных подпространств. Базовые алгоритмы обучаются на различных подмножествах признаков, которые выделяются случайным образом.

Преимущества случайного леса:

* высокая точность предсказания;
* редко переобучается;
* практически не чувствителен к выбросам в данных;
* одинаково хорошо обрабатывает как непрерывные, так и дискретные признаки, данные с большим числом признаков;
* высокая параллелизуемость и масштабируемость.

Из недостатков можно отметить, что его построение занимает больше времени. Так же теряется интерпретируемость.

* + 1. **Градиентный бустинг (Gradient Boosting)**

Градиентный бустинг — еще один представитель ансамблевых методов.

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

Чтобы построить алгоритм градиентного бустинга, нам необходимо выбрать базовый алгоритм и функцию потерь или ошибки (loss). Loss-функция

– это мера, которая показывает насколько хорошо предсказание модели соответствуют данным. Используя градиентный спуск и обновляя предсказания, основанные на скорости обучения (learning rate), ищем значения, на которых loss минимальна.

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

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

* + 1. **Нейронная сеть**

Нейронная сеть — это последовательность нейронов, соединенных между собой связями. Структура нейронной сети пришла в мир программирования из биологии. Вычислительная единица нейронной сети — нейрон или персептрон.

У каждого нейрона есть определённое количество входов, куда поступают сигналы, которые суммируются с учётом значимости (веса) каждого входа.

Смещение – это дополнительный вход для нейрона, который всегда равен 1 и, следовательно, имеет собственный вес соединения.

Так же у нейрона есть функция активации, которая определяет выходное значение нейрона. Она используется для того, чтобы ввести нелинейность в нейронную сеть. Примеры активационных функций: relu, сигмоида, tanh и др.

У полносвязной нейросети выход каждого нейрона подается на вход всем нейронам следующего слоя. У нейросети имеется:

* входной слой — его размер соответствует входным параметрам;
* скрытые слои — их количество и размерность определяем специалист;
* выходной слой — его размер соответствует выходным параметрам.

Прямое распространение – это процесс передачи входных значений в нейронную сеть и получения выходных данных, которые называются прогнозируемым значением.

Прогнозируемое значение сравниваем с фактическим с помощью функции потери. В методе обратного распространения ошибки градиенты (производные значений ошибок) вычисляются по значениям весов в направлении, обратном прямому распространению сигналов. Значение градиента вычитают из значения веса, чтобы уменьшить значение ошибки. Таким образом происходит процесс обучения. Обновляются веса каждого соединения, чтобы функция потерь минимизировалась.

Для обновления весов в модели используются различные оптимизаторы.

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

Нейронные сети применяются для решения задач регрессии, классификации, распознавания образов и речи, компьютерного зрения и других. На настоящий момент это самый мощный, гибкий и широко применяемый инструмент в машинном обучении.

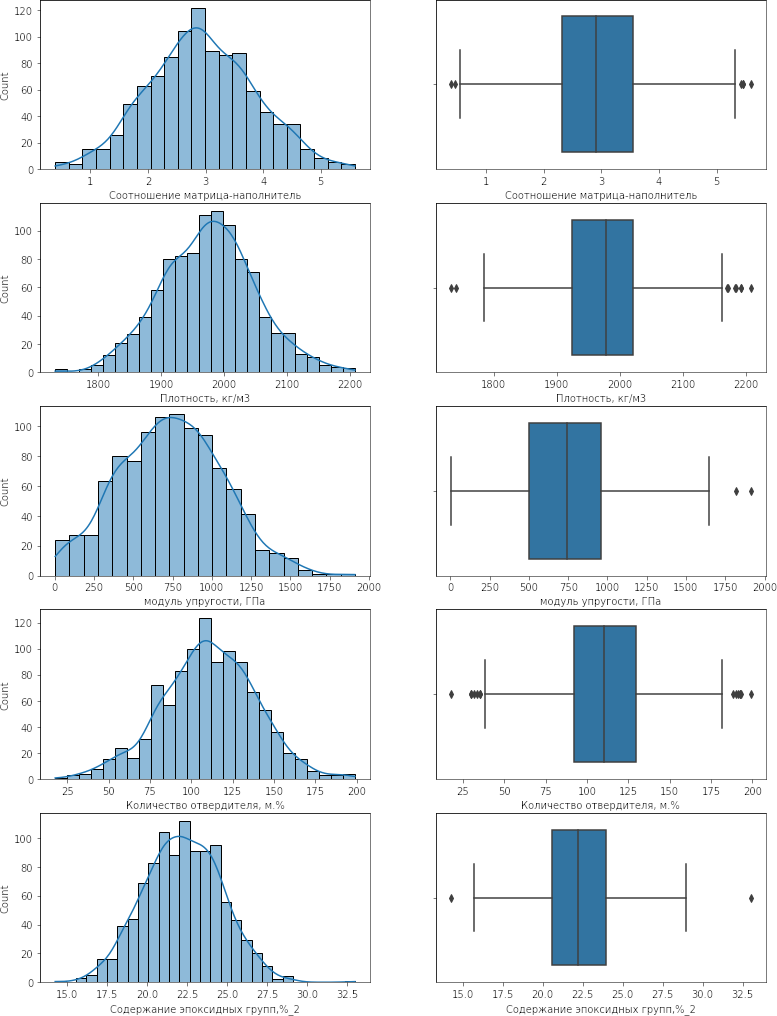
**1.3 Разведочный анализ данных**

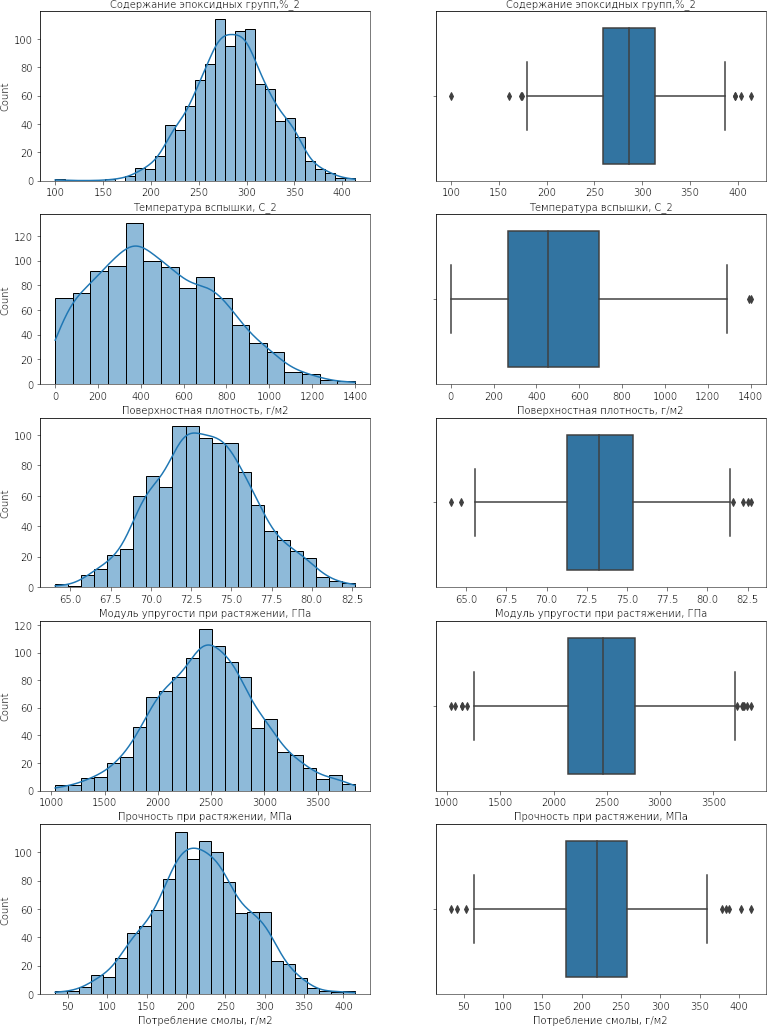
Описание признаков объединенного датасета приведено в таблице 1. Все признаки имеют тип данных float64, за исключением признака «Угол нашивки» (который имеет тип int64 в исходных данных). Пропусков в данных нет. Все признаки, кроме «Угол нашивки», являются непрерывными. «Угол нашивки» принимает только два значения.

**Таблица 1. Описание признаков датасета**

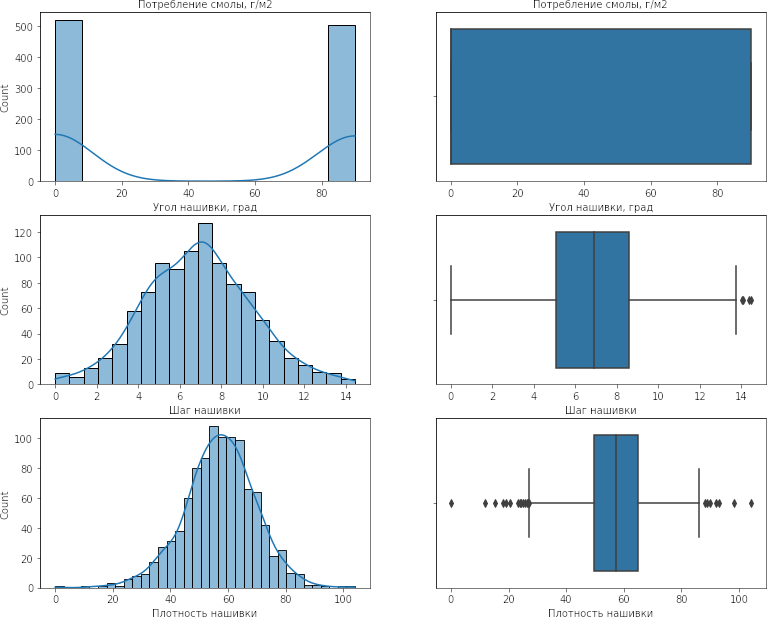
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название | Файл | Тип данных | Непустых значений | Уникальных значений |
| Соотношение матрица- наполнитель | X\_bp | float64 | 1023 | 1014 |
| Плотность, кг/м3 | X\_bp | float64 | 1023 | 1013 |
| модуль упругости, ГПа | X\_bp | float64 | 1023 | 1020 |
| Количество отвердителя, м.% | X\_bp | float64 | 1023 | 1005 |
| Содержание эпоксидных групп,%\_2 | X\_bp | float64 | 1023 | 1004 |
| Температура вспышки, С\_2 | X\_bp | float64 | 1023 | 1003 |
| Поверхностная плотность, г/м2 | X\_bp | float64 | 1023 | 1004 |
| Модуль упругости при растяжении, ГПа | X\_bp | float64 | 1023 | 1004 |
| Прочность при растяжении, МПа | X\_bp | float64 | 1023 | 1004 |
| Потребление смолы, г/м2 | X\_bp | float64 | 1023 | 1003 |
| Угол нашивки, град | X\_nup | intt64 | 1023 | 2 |
| Шаг нашивки | X\_nup | float64 | 1023 | 989 |
| Плотность нашивки | X\_nup | float64 | 1023 | 988 |

Гистограммы распределения переменных и диаграммы «ящик с усами» приведены на рисунке 1. Отметим, что почти все признаки имеют близкое к нормальному распределения. Исключением являются: признак «Угол нашивки», принимающим всего два значения 0 и 90 град., и «Поверхностная плотность», который имеет скошенное распределение.



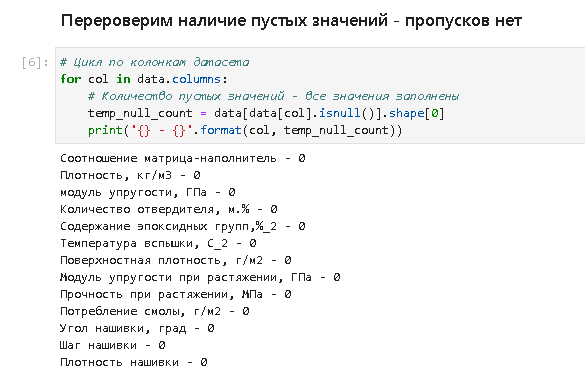


**Рисунок 1. Гистограммы распределения и диаграммы «ящик с усами» для каждого признака**



**Рисунок 1 (продолжение). Гистограммы распределения и диаграммы «ящик с усами» для каждого признака**

Пропуски в данных не обнаружены (см рис. 2).

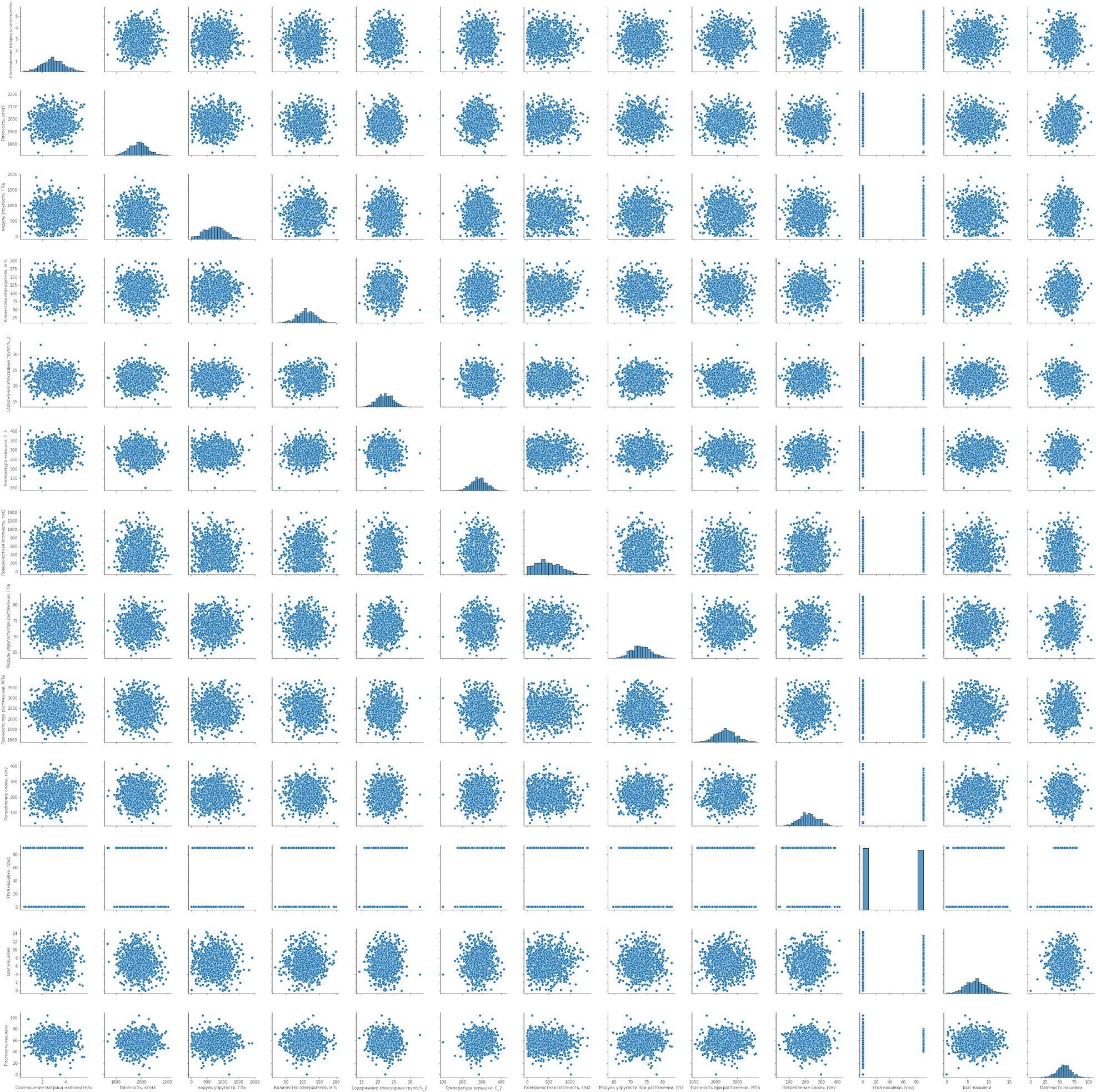


**Рисунок 2. Проверка пропусков в признаках**

Описательная статистика представлена в таблице 2.

**Таблица 2 — Описательная статистика признаков**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Среднее | Стандартное  отклонение | Минимум | Максимум | Медиана |
| Соотношение  матрица-наполнитель | 2.9304 | 0.9132 | 0.3894 | 5.5917 | 2.9069 |
| Плотность, кг/м3 | 1975.7349 | 73.7292 | 1731.7646 | 2207.7735 | 1977.6217 |
| модуль упругости,  ГПа | 739.9232 | 330.2316 | 2.4369 | 1911.5365 | 739.6643 |
| Количество  отвердителя, м.% | 110.5708 | 28.2959 | 17.7403 | 198.9532 | 110.5648 |
| Содержание эпоксидных групп,  %\_2 | 22.2444 | 2.4063 | 14.2550 | 33.0000 | 22.2307 |
| Температура  вспышки, С\_2 | 285.8822 | 40.9433 | 100.0000 | 413.2734 | 285.8968 |
| Поверхностная  плотность, г/м2 | 482.7318 | 281.3147 | 0.6037 | 1399.5424 | 451.8644 |
| Модуль упругости при растяжении, ГПа | 73.3286 | 3.1190 | 64.0541 | 82.6821 | 73.2688 |
| Прочность при  растяжении, МПа | 2466.9228 | 485.6280 | 1036.8566 | 3848.4367 | 2459.5245 |
| Потребление смолы,  г/м2 | 218.4231 | 59.7359 | 33.8030 | 414.5906 | 219.1989 |
| Угол нашивки, град | 44.2522 | 45.0158 | 0.0000 | 90.0000 | 0.0000 |
| Шаг нашивки | 6.8992 | 2.5635 | 0.0000 | 14.4405 | 6.9161 |
| Плотность нашивки | 57.1539 | 12.3510 | 0.0000 | 103.9889 | 57.3419 |



**Рисунок 3. Попарные графики рассеяния точек**

Попарные графики рассеяния точек приведены на рисунке 3.

По графикам рассеяния очевидно отсутствие линейной зависимости между

признаками.

На рис. 1 определяются выбросы в признаках на графиках «ящик с усами».

В данной работе уделено большое внимание обработке выбросов. В приложенных к данной записке «ноутбуке» № 4 проработаны следующие методы определения и обработки выбросов:

* определение выбросов в признаках с распределением близким к

нормальному:

- на основе правила «3-х сигм»;

- по квантилям (использовались 5 %- 95% квантили и 2.5% - 97.5 %)

* определение выбросов в признаках со скошенным распределением:

- на основе межквартильного размаха (коэффициент К = 1.5);

* обработка выбросов, определенных вышеописанными способами, проводилась:

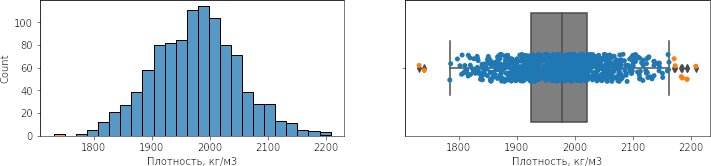
- удалением;

- заменой граничными значениями («хвостами распределений»);

- заменой центрами распределений (по моде в признаке со скошенным распределением и по медиане во всех остальных).

Отметим, что в признаке «Угол нашивки» определение выбросов не проводилось ввиду бинарности признака.

На рисунке 4 приведен пример наложения точек распределения признака на график «ящик с усами» (boxplot).



**Рисунок 4. Наложение распределения на график boxplot**

В «ноутбуке» № 4 проведен эксперимент, в ходе которого было показано, что рассмотренные методы обработки выбросов не существенно влияют на качество моделей по метрике МАЕ (разница составила 0.01). На этом основании, на дальнейшую предобработку подавались оригинальные данные, не обработанные на выбросы.

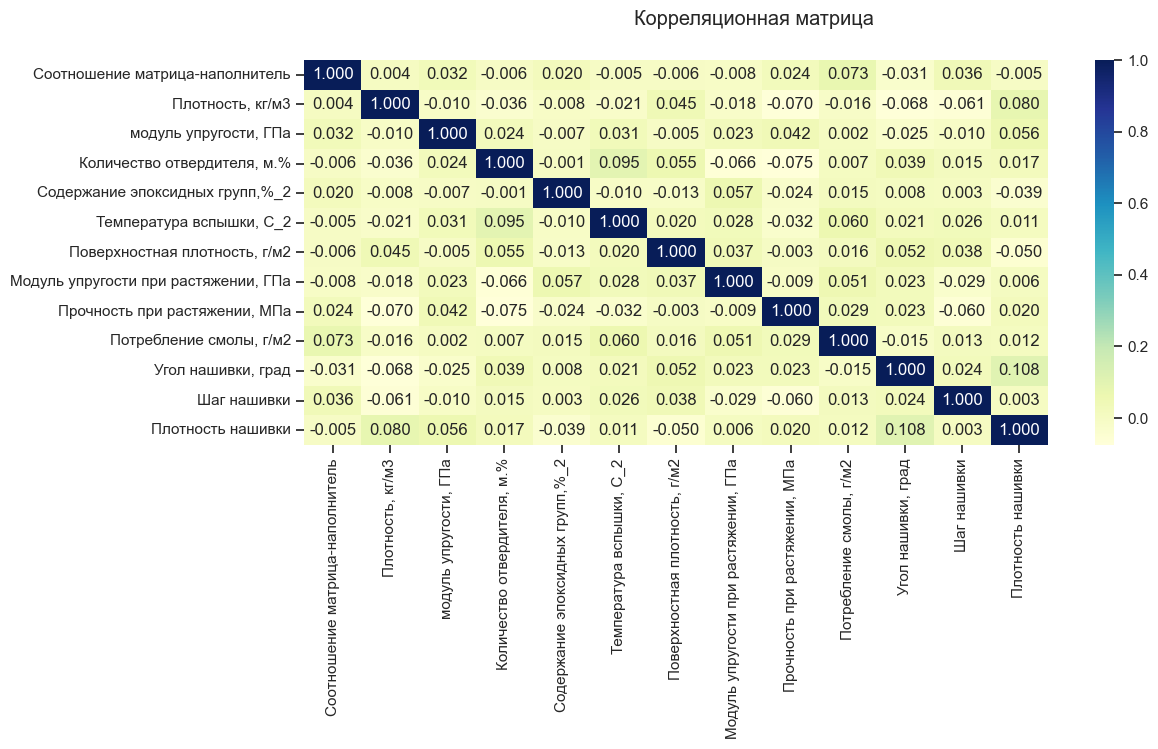
Одной из целей разведочного анализа данных — выявить закономерности в данных. Для корректной работы большинства моделей желательна существенная зависимость целевых признаков от входных и отсутствие зависимости между входными признаками. Это позволяет понижать размерность поля входных признаков и добавлять в данные новые полезные признаки, т.е. заниматься Feature Generation и Feature Selection.

На рисунке 3 мы видели график попарного рассеяния точек. По форме

«облаков точек» мы не заметили зависимостей, которые станут основой работы моделей. Помочь выявить связь между признаками может матрица корреляции Пирсона, приведенная на рисунке 5 (см. на следующей стр.).

По матрице корреляции мы видим, что все коэффициенты корреляции близки к нулю, что означает отсутствие линейной зависимости между признаками.

Разведочный анализ проведен в «ноутбуке» № 2.



**Рисунок 5. Матрица корреляции**

# Практическая часть

* 1. **Предобработка данных.**

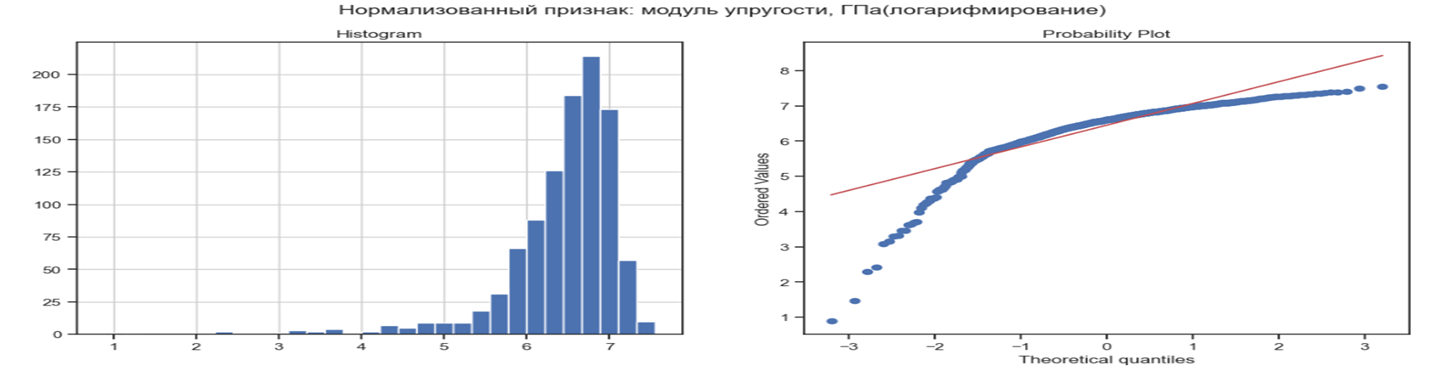
Предобработка данных выполнялась в следующем порядке:

1. Нормализация признаков;
2. Обработка выбросов;
3. Масштабирование (стандартизация).

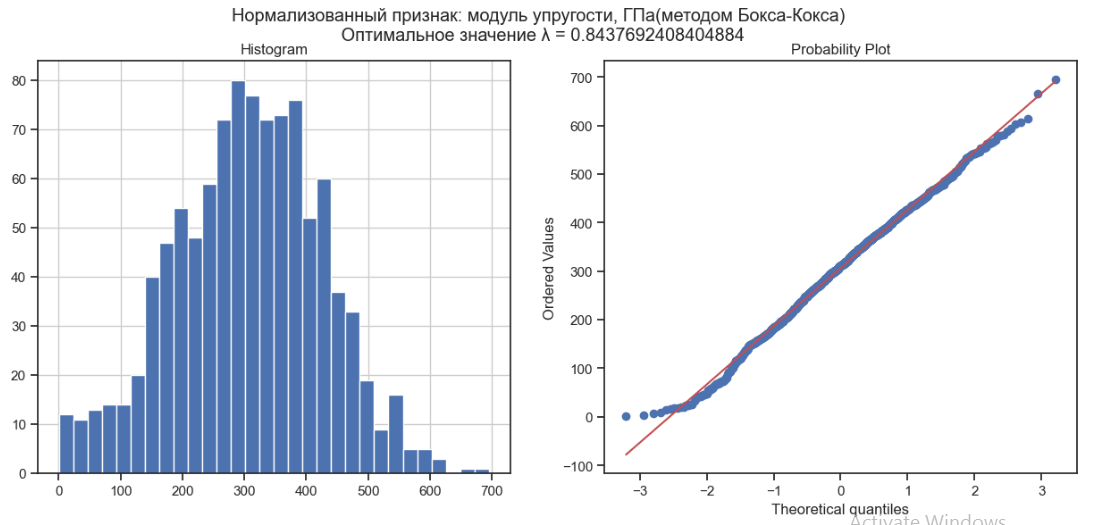
**2.1.1 Нормализация признаков**

Выполнена в «ноутбуке» № 3. Рассматривалось три варианта нормализации признаков, один «классический» - логарифмирование, и два специализированных: метод Бокса-Кокса и Йео-Джонсона.

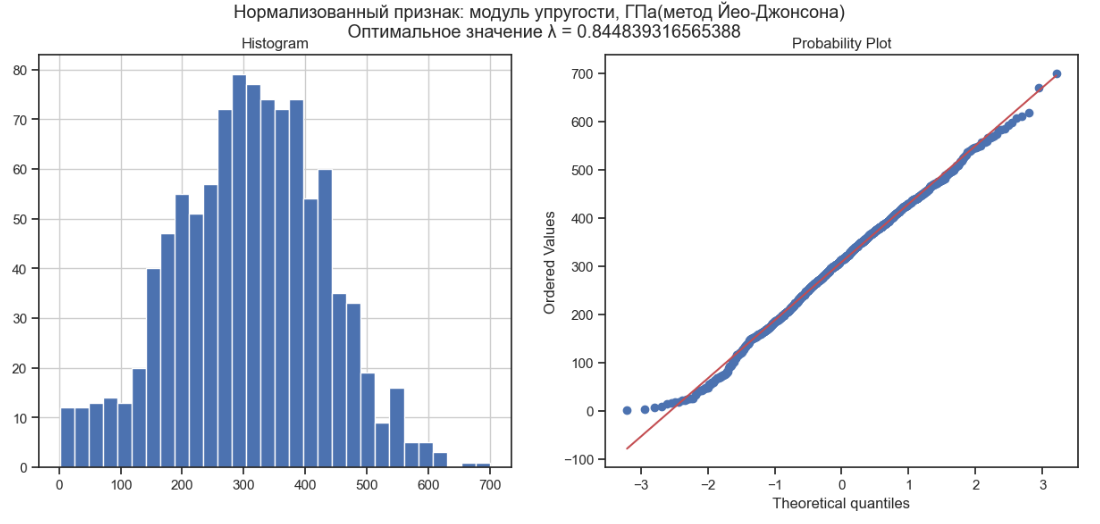
На рисунках 6, 7 и 8 представлены примеры нормализации указанными способами. Как видно на графике «Квантиль – квантиль» (Q-Q plot), специализированные методы дают лучшие результаты.



**Рисунок 6. Нормализация логарифмированием.**



**Рисунок 7. Нормализация методом Бокса-Кокса.**



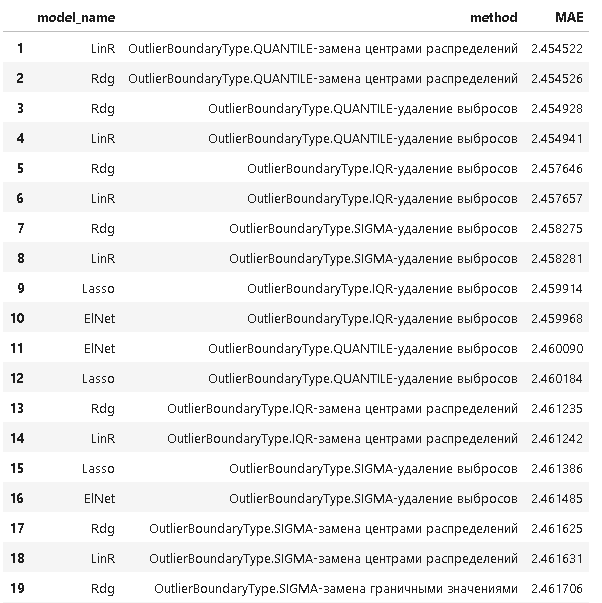
**Рисунок 8. Нормализация методом Йео-Джонсона.**

Выбор сделан в пользу метода Йео-Джонсона, т.к. он позволяет обрабатывать признаки содержащие нули, которые присутствуют в датасете.

**2.1.2 Обработка выбросов**

К сказанному в разделе 1.3 об обработке выбросов, проделанной в данной работе, здесь добавим результаты проведенного эксперимента в «ноутбуке» № 4. Напомним, что суть эксперимента – определить, оказывает ли существенное влияние обработка выбросов предложенными способами на качество моделей.

На рисунке 9, частично приведена таблица результатов эксперимента. Метрикой качества выбрана МАЕ. По ней видно, что оригинальные данные, т.е. где обработка выбросов не проводилась, расположились на 21, 22 позициях, имея показатель МАЕ лишь на 0.01 хуже, чем обработанные данные (заменой центрами распределений, удалением выбросов). Таким образом, рассмотренные методы обработки выбросов не существенно влияют на качество моделей.



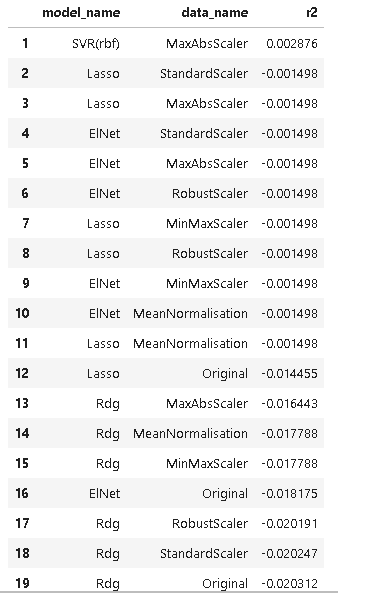


**Рисунок 9. Результаты эксперимента влияния обработки выбросов.**

**2.1.3 Масштабирование (стандартизация).**

В «ноутбуке» № 5 был проведен эксперимент, целью которого было определить, влияет ли выбор метода масштабирования на качество моделей. В этом эксперименте метрикой качества моделей взят коэффициент детерминации (R2).

На рисунке 10 частично приведена таблица с результатами. Как видно, произошла градация моделей по метрике R2 после масштабирования разными способами, а не самих методов масштабирования. Что говорит, о пригодности всех рассмотренных методов масштабирования.



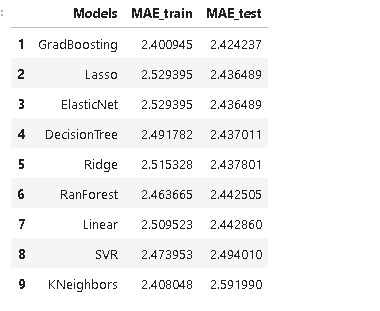
**Рисунок 10. Результаты эксперимента влияния масштабаторов.**

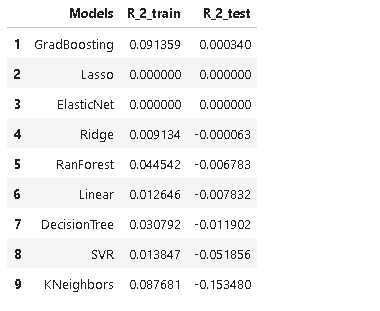
В работе, в «ноутбуках» использовались StandardScaler, RobustScaler и MaxAbsoluteScaler из библиотеки Sklearn.

* 1. **Разработка и обучение моделей.**

Так как задано три целевых признака, подбирались и обучались 3 модели. Выбор моделей проходил вместе с подбором гипперпараметров к ним и кросс-валидации (число фолдов равно 10). Для «Модуля упругости при растяжении» рассматривались все 9 моделей, описанных выше. Для «Прочности при растяжении» отбросили 4 модели, качество работы которых, вызывало сомнения.

На рисунках 11, 12 приведены сводные таблицы моделей по метрикам качества МАЕ и R2 после подбора гипперпараметров при кроссвалидации выполненной при помощи класса GridSearchCV из библиотеки Sklearn.



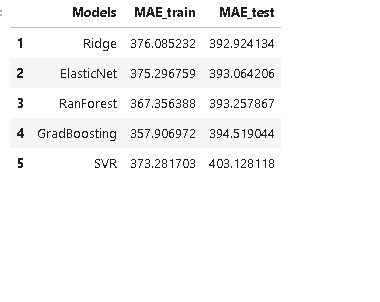


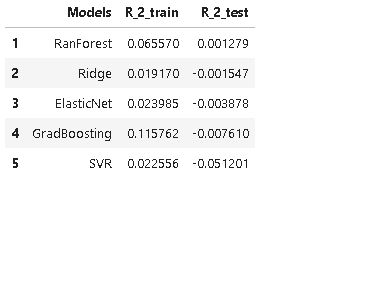
**Рисунок 11. Метрики качества моделей прогнозирования признака «Модуль упругости при растяжении».**

Для прогнозирования «Модуля упругости при растяжении» была выбрана модель Ridge Regressor.

Для прогнозирования «Прочности при растяжении» - Elastic Net Regressor.

Эти модели показали наилучшие результаты на тестовой выборке.

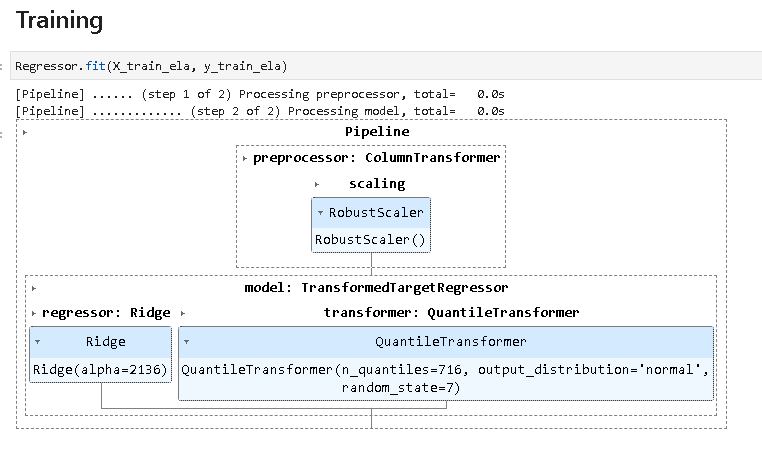




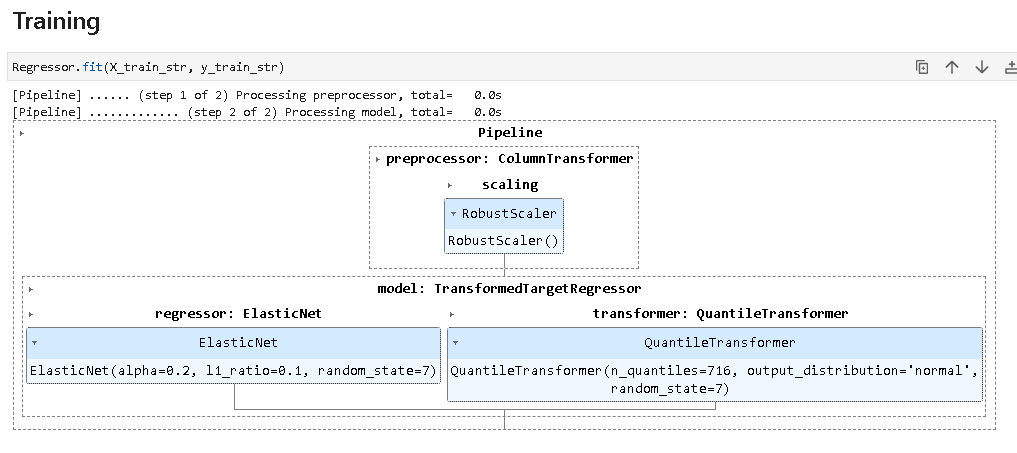
**Рисунок 12. Метрики качества моделей прогнозирования признака «Прочность при растяжении».**

На рисунках 13, 14 приводится схема «собранных» моделей для задач прогнозирования «Модуля упругости при растяжении» и «Прочности при растяжении».

Лучший результат выбранные модели показали с данными, предобработанными RobustScaler (масштабирование по медиане).



**Рисунок 13. Схема модели** **прогнозирования признака «Модуль упругости при растяжении».**



**Рисунок 14. Схема модели** **прогнозирования признака «Прочность при растяжении».**

**2.3 Тестирование моделей.**

На рисунках 15, 16 приведены результаты предсказания моделей на тестовой выборке с оценкой результатов качества.

Как видно по метрике R2, отобранные модели обучились предсказывать среднее или близкое к среднему значение целевых признаков по тестовой выборке на все входные комбинации признаков-предикторов.

Метрика МАЕ, выраженная в относительных цифрах от размаха значений целевых признаков тестовой выборке, составляет примерно 14 %.

Средняя квадратическая ошибка предсказаний (посчитанная на рисунках как квадратный корень из MSE) почти равна стандартному отклонению целевых признаков на тестовой выборке.



**Рисунок 15. Оценка качества модели («Модуль упругости при растяжении»).**



**Рисунок 16. Оценка качества модели («Прочность при растяжении»)**

Модели прогноза целевых признаков «Модуль упругости при растяжении» и «Прочность при растяжении» представлены в «ноутбуках» № 7.1 и 7.2 соответственно.

**2.4 Нейронная сеть.**

Архитектура нейронной сети для предсказания целевого признака «Матрица-наполнитель» подбиралась с помощью фреймворка подбора гипперпараметров Keras Tuner из библиотеки Keras. Рассматривались два варианта – с двумя и тремя полносвязными скрытыми слоями с добавлением слоев «дропаут» для решения проблемы переобучения моделей (см. «ноутбуки» 6.3.1 и 6.3.2).

Реализованы оба варианта многослойного персептрона, соответственно, в «ноутбуках» 7.3.1 и 7.3.2.

На рисунке 17 представлен код с подбираемыми гипперпараметрами по которым Keras Tuner выбрал наилучшую комбинацию для персептрона с тремя полносвязными скрытыми слоями.



**Рисунок 17. Подбор гипперпараметров нейросети**

На рисунке 18 представлена архитектура этой нейросети с подобранными Keras Tuner гипперпараметрами.



**Рисунок 18. Нейросеть с подобранными Keras Tuner гипперпараметрами.**

На рисунке 19 показана оценка качества нейросети с тремя скрытыми полносвязными слоями реализованной в «ноутбуке» 7.3.2.

Нейросеть по метрикам качества R2, MAE и MSE в целом дала похожий результат в сравнении с выше рассмотренными моделями – сеть предсказывает значение близкое к среднему целевого признака на тестовой выборке на все входящие комбинации признаков предикторов.



**Рисунок 19. Оценка качества модели («Соотношение матрица-наполнитель»)**

Метрика МАЕ, выраженная в относительных цифрах от размаха значений целевого признака тестовой выборке, составляет примерно 13.5 %.

Средняя квадратическая ошибка предсказаний (посчитанная на рисунках как квадратный корень из MSE) почти равна стандартному отклонению целевых признаков на тестовой выборке.

**2.5 Разработка приложения.**

Разработка веб-приложения на языке Python с помощью фреймворка Flask и шаблонизатора Jinja.

В приложении планируется реализовать реализовать следующие функции:

* выбор целевой переменной для предсказания;
* ввод входных параметров;
* проверка введенных параметров;
* загрузка сохраненной модели, получение и отображение прогноза целевых признаков.

**2.6 Создание удаленного репозитория.**

Данная работа размещена на репозитории GitHub, который находится по адресу https://github.com/\_\_\_\_\_\_\_\_\_\_.

**Заключение.**

Подведем итоги работы. Модели обучились предсказывать среднее значение целевых признаков на тестовой выборке. Считать такой результат удовлетворительным нельзя.

Вероятные причины:

* Недостаточность датасета по объему данных;
* Отсутствие значимой корреляции между признаками не дает явных оснований генерировать новые признаки, отсеивать «вредные» для улучшения качества прогноза;
* Отсутствие достаточного изучения природы данных, предметной области для поиска и генерации не выявленных взаимосвязей между признаками.

.

# Библиографический список.

1. Композиционные материалы : учебное пособие для вузов / Д. А. Иванов, А. И. Ситников, С. Д. Шляпин ; под редакцией А. А. Ильина. — Москва : Из- дательство Юрайт, 2019 — 253 с. — (Высшее образование). — Текст : непо- средственный.
2. Силен Дэви, Мейсман Арно, Али Мохамед. Основы Data Science и Big Data. Python и наука о данных. – СПб.: Питер, 2017. – 336 с.: ил.
3. ГрасД. Data Science. Наука о данных с нуля: Пер. с англ. - 2-е изд., пере- раб. и доп. - СПб.: БХВ-Петербурr, 2021. - 416 с.: ил.
4. Документация по языку программирования python: – Режим доступа: https://docs.python.org/3.8/index.html.
5. Документация по библиотеке numpy: – Режим доступа: https:// numpy.org/doc/1.22/user/index.html#user.
6. Документация по библиотеке pandas: – Режим доступа: https:// pandas.pydata.org/docs/user\_guide/index.html#user-guide.
7. Документация по библиотеке matplotlib: – Режим доступа: https:// matplotlib.org/stable/users/index.html.
8. Документация по библиотеке seaborn: – Режим доступа: https:// seaborn.pydata.org/tutorial.html.
9. Документация по библиотеке sklearn: – Режим доступа: https://scikit- learn.org/stable/user\_guide.html.
10. Документация по библиотеке keras: – Режим доступа: https://keras.io/

api/.

1. Руководство по быстрому старту в flask: – Режим доступа: https://flask-

russian-docs.readthedocs.io/ru/latest/quickstart.html.

1. Loginom Вики. Алгоритмы: – Режим доступа: https://wiki.loginom.ru/ algorithms.html.
2. Andre Ye. 5 алгоритмов регрессии в машинном обучении, о которых вам следует знать: – Режим доступа:https://habr.com/ru/company/vk/blog/513842/. 14 Alex Maszański. Метод k-ближайших соседей (k-nearest neighbour): – Режим доступа: https://proglib.io/p/metod-k-blizhayshih-sosedey-k-nearest-

neighbour.

1. Yury Kashnitsky. Открытый курс машинного обучения. Тема 3. Классификация, деревья решений и метод ближайших соседей: – Режим до- ступа: https://habr.com/ru/company/ods/blog/322534/.
2. Yury Kashnitsky. Открытый курс машинного обучения. Тема 5. Компо- зиции: бэггинг, случайный лес: – Режим доступа: https://habr.com/ru/company/ ods/blog/324402/.
3. Alex Maszański. Машинное обучение для начинающих: алгоритм слу- чайного леса (Random Forest): – Режим доступа: https://proglib.io/p/mashinnoe- obuchenie-dlya-nachinayushchih-algoritm-sluchaynogo-lesa-random-forest.
4. Alex Maszański. Решаем задачи машинного обучения с помощью алго- ритма градиентного бустинга: – Режим доступа: https://proglib.io/p/reshaem- zadachi-mashinnogo-obucheniya-s-pomoshchyu-algoritma-gradientnogo-bustinga.