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 Pro»

**Тема:**

Прогнозирование конечных свойств новых материалов (композиционных материалов).

Слушатель Маслов Виктор Николаевич

Москва, 2025

**Содержание**

**Введение…………………………………………………………………………..3**

**Аналитическая часть……………………………………………………………5**

Постановка задачи………………………………………………………………...5

Характеристика дата сета…………………………………………………………5

Описание используемых методов ……………………………….……………….7

Линейная регрессия (полиномиальная регрессия)………………………9

Случайный лес .………………………………………………………….11

Градиентный бустинг……………………………………………………13

Нейроная сеть………………………………………………………….…14

Условия работоспособности применяемых методов…………………………..16

Разведочный анализ данных (EDA)……………………………………………..17

Инструменты и методы (EDA)…………………………………………………..18

Описание данных………………………………………………………….18

Визуализация данных……………………………………………………..19

Сводные статистики………………………………………………………19

Преобразование данных…………………………………………………..20

**Практическая часть……………………………………………………………21**

Предобработка данных…………………………………………………………..21

Очистка данных от выбросов……………………………………………………25

Разработка и обучение модели…………………………………………………..26

Тестирование модели ……………………………………………………………26

Смена постановки задачи………………………………………………………..29

Описание используемых методов………………………………………………29

К-средних……………………………………………………………….....29

К-ближайших соседей…………………………………………………….30

Случайный лес………………………………………...…………………..32

Разработка и тестирование модели.………………………………………….…33

Нейросети.………………………………………………………………………..35

Создание удаленного репозитория……………………………………………..36

**Литература………………………………………………………………………37**

**Введение**

Наука о данных - это передовая и быстроразвивающаяся отрасль знаний. Она находится на пересечении: алгоритмических навыков, знаний математики и статистики, профессионального опыта в предметной области.

Исследователь данных (Data Scientist), как настоящий учёный, занимается не только сбором и анализом данных, но и изучает их в разных контекстах и под разными углами, подвергая сомнению любые предположения: статистические методы, моделирование баз данных, методы интеллектуального анализа приложения искусственного интеллекта для работы с данными, методы проектирования и разработки баз данных.

В выпускной квалификационной работе перед нами стояла задача прогнозирования – создание прогнозных моделей. Прогнозирование является важной задачей в различных областях, таких как экономика, финансы, маркетинг и другие. Методы прогнозирования позволяют предсказывать будущие значения переменных или каких-либо свойств, на основе имеющихся данных. Одним из наиболее эффективных методов прогнозирования является метод машинного обучения.

Метод машинного обучения – это подход к анализу данных, при котором компьютерные системы обучаются на основе опыта и самостоятельно улучшают свою производительность без явного программирования. В контексте прогнозирования, модели машинного обучения используют имеющиеся данные для построения математической модели, способной предсказывать будущие значения.

Существует несколько основных типов моделей машинного обучения для прогнозирования. Один из них - линейная регрессия. Линейная регрессия предполагает линейную зависимость между зависимой переменной и одной или несколькими независимыми переменными. Модель строит линию или поверхность, которая наилучшим образом соответствует имеющимся данным. Используя эту линию или поверхность, можно предсказать будущие значения зависимой переменной на основе значений независимых переменных.

Другой тип модели машинного обучения для прогнозирования - деревья решений. Деревья решений строятся в виде иерархических структур, где каждый узел представляет собой вопрос о значении определенной переменной. Ответы на эти вопросы определяют, по какому пути следовать по дереву для получения прогноза. Этот тип модели особенно полезен при работе с категориальными или бинарными переменными.

Кроме того, существуют и другие методы машинного обучения для прогнозирования, такие как нейронные сети и алгоритмы кластеризации. Нейронные сети имитируют работу человеческого мозга и состоят из множества связанных нейронов, которые передают информацию друг другу. Алгоритмы кластеризации позволяют группировать данные на основе их сходства или различия.

Однако, необходимо учитывать, что модели машинного обучения не всегда дают точные прогнозы. Важно правильно выбрать модель и обучить ее на достаточном количестве данных для достижения наилучшей производительности. Также стоит отметить, что прогнозирование может быть сложной задачей из-за нестабильности данных или изменения условий в будущем.

В заключение, методы машинного обучения предоставляют мощный инструмент для прогнозирования будущих значений переменных. Линейная регрессия, деревья решений, нейронные сети и алгоритмы кластеризации - все они предлагают различные подходы к прогнозированию и имеют свои особенности. Правильный выбор модели и оптимальное использование имеющихся данных помогут достичь наилучших результатов при прогнозировании методом машинного обучения

**1.Аналитическая часть**

* 1. **Постановка задачи**

Правильная постановка задачи в машинном обучении является основой успешного проекта. Она помогает определить направление работы, выбрать подходящие методы и оценить результаты, что в конечном итоге приводит к более эффективным решениям.

Выпускная квалификационная работа основана на реальных производственных задачах Центра НТИ «Цифровое материаловедение: новые материалы и вещества» и состоит в создание прогнозной модели. Суть прогнозирования заключается в симуляции представительного элемента объема композита на основе данных о характеристиках входящих компонентов (связующего и армирующего компонента).

На входе имеются данные о начальных свойствах компонентов композиционных материалов. На выходе необходимо спрогнозировать ряд конечных свойств получаемых композиционных материалов.

Исходя из вышесказанного перед нами стоит задача регрессии:

- предсказать неизвестные значения зависимых переменных;

- выявить, какие независимые переменные связаны с зависимыми;

- понять отношения между зависимыми и независимыми переменными.

* + 1. **Характеристика датасета**

Входящие данные состоят из двух файлов формата «\*.xlsx», которые были загружены в «Jupyter Noutbook» и объединены методом «join», используемый библиотекой «pandas», по индексу. Тип объединения «inner». Ниже, на Таблице 1, приведена характеристика объеденного датасета.

Таблица 1-Хараткеристика датасета

|  |  |  |
| --- | --- | --- |
| **Описание датасета** | | |
| Название и источник данных | Данные предоставлены администрацией ЦДО МГТУ им. Н.Э.Баумана. Файлы: «X\_bp.xlsx», «X\_nup.xlsx» | |
| **Структура данных** | | |
| Формат данных | Microsoft Excel | |
| Количество строк | 1023 | |
| Количество столбцов | 13 | |
| **Признаки (фичи)** | | |
| Список признаков (тип) | Соотношение матрица-наполнитель | float64 |
| Плотность, кг/м3 | float64 |
| Модуль упругости, ГПа | float64 |
| Количество отвердителя, м.% | float64 |
| Содержание эпоксидных групп,%\_2 | float64 |
| Температура вспышки, С\_2 | float64 |
| Поверхностная плотность, г/м2 | float64 |
| Потребление смолы, г/м2 | float64 |
| Угол нашивки, град | int |
| Шаг нашивки | float64 |
| Плотность нашивки | float64 |
| **Качество данных** | | |
| Пропуски | Отсутствуют | |
| Выбросы | Количество выбросов находиться в диапазоне от 2-21. | |
| **Переменные** | | |
| Вход | 13 | |
| Выход | 2 | |

* 1. **Описание используемых методов**

Методы решения задач машинного обучения можно классифицировать по различным критериям, включая тип задачи, используемые алгоритмы и подходы к обучению.

В задаче регрессии используются методы, позволяющие вычислить предполагаемые отношения между зависимой и независимыми переменными. Используя регрессионный анализ, мы можем моделировать отношения между выбранным переменными, а также прогнозировать значениями на основе модели.

Регрессионный анализ использует заданный метод оценки, зависимую переменную и одну или несколько независимых переменных для создания уравнения, которое оценивает значения зависимой переменной.

Методы решения задач регрессии направлены на предсказание непрерывных значений на основе входных данных. Некоторые методы, которые используются для решения задач регрессии приведены в Таблице 2.

Таблица 2 – Основные методы задач регрессии

|  |  |
| --- | --- |
| **Метод** | **Описание** |
| Линейная регрессия | Базовый метод, который предполагает линейную зависимость между независимыми переменными (признаками) и зависимой переменной (целевым значением). Модель строится с помощью минимизации суммы квадратов ошибок. |
| Полиномиальная регрессия | Расширяет линейную регрессию, позволяя моделировать нелинейные зависимости. Она включает в себя полиномиальные термины (например, квадратные или кубические), что позволяет лучше подстраиваться под сложные зависимости в данных. |

Продолжение Таблицы 2

|  |  |
| --- | --- |
| **Метод** | **Описание** |
| Деревьев решений | Деревья решений могут быть использованы для регрессии, где каждое решение (узел) делит данные на подгруппы на основе значений признаков. Это позволяет моделировать сложные зависимости и взаимодействия между признаками. |
| Случайный лес | Ансамблевый метод, который использует множество деревьев решений для улучшения точности предсказаний. Он уменьшает риск переобучения и повышает устойчивость модели за счет усреднения предсказаний нескольких деревьев. |
| Градиентный бустинг | Градиентный бустинг строит модели последовательно, каждая новая модель исправляет ошибки предыдущей. Этот метод часто показывает высокую точность и может быть использован для решения как задач классификации, так и регрессии. |
| Регрессия с опорными векторами (SVR) | Методы опорных векторов могут быть адаптированы для задач регрессии. SVR пытается найти гиперплоскость, которая минимизирует ошибки предсказания, при этом допускает определенное количество ошибок в пределах заданного порога. |
| Нейронные сети | Нейронные сети, особенно глубокие, могут быть использованы для регрессии, позволяя моделировать сложные и нелинейные зависимости. Они хорошо работают с большими объемами данных и могут адаптироваться к различным типам задач. |

Выбор метода регрессии зависит от характера данных, сложности задачи и требований к интерпретируемости модели. Правильный выбор алгоритма и его параметров может значительно повлиять на качество предсказаний и успешность проекта в области машинного обучения.

Рассмотрим подробнее модели, которые планируем использовать в нашей работе.

**Линейная регрессия (полиноминальная регрессия)**

В машинном и глубоком обучении линейная регрессия занимает особое место, являясь не просто статистическим инструментом, но, а также фундаментальным компонентом для многих более сложных концепций.

Линейная регрессия, описывает зависимость целевой переменной от признака в виде линейной функции: y = kx + b. Уравнение вида: fw,b(x) = w0x0 + w1x1 +... + wnxn + b = w x + b называется множественной линейной регрессией, где b — смещение модели, w — вектор её весов, а x — вектор признаков одного обучающего образца.

Линейную регрессию также можно применять к данным с нелинейной зависимостью, добавив степени каждого признака в виде новых признаков с последующим обучением на полученном датасете.

Сильные стороны метода:

1. Простота и интерпретируемость: Линейная регрессия легко понимается и интерпретируется. Коэффициенты модели показывают, как изменение независимой переменной влияет на зависимую.
2. Быстрота обучения: Модель линейной регрессии обучается быстро, даже на больших наборах данных.
3. Малое количество параметров: Линейная регрессия требует меньше параметров по сравнению с более сложными моделями, что снижает риск переобучения.
4. Эффективность: Для линейных зависимостей линейная регрессия может быть очень эффективной и давать хорошие результаты.
5. Статистические предположения: Линейная регрессия позволяет проводить статистические тесты, такие как тесты значимости коэффициентов.

Слабые стороны метода:

1. Линейность: Линейная регрессия предполагает, что зависимость между переменными линейна. Если данные имеют нелинейные зависимости, модель может не дать хороших результатов.
2. Чувствительность к выбросам: Линейная регрессия чувствительна к выбросам, которые могут значительно исказить результаты.
3. Мультиколлинеарность: Если независимые переменные сильно коррелируют друг с другом, это может привести к нестабильности коэффициентов и затруднить интерпретацию модели.
4. Предположения о нормальности: Линейная регрессия предполагает, что ошибки распределены нормально. Если это предположение нарушается, результаты могут быть ненадежными.
5. Ограниченная гибкость: Линейная регрессия не может захватывать сложные зависимости и взаимодействия между переменными без дополнительных преобразований или добавления полиномиальных признаков.

Линейная регрессия является мощным инструментом для анализа данных, но её применение должно быть обосновано. Важно проверять предположения модели и рассматривать альтернативные методы, если данные не соответствуют этим предположениям.

**Случайный лес**

Случайный лес — это ансамблевый метод машинного обучения, который использует множество деревьев решений для улучшения точности и устойчивости модели. Он сочетает в себе идеи бутстрэппинга (статистический метод, который используется для оценки распределения выборочных статистик путем многократного случайного выбора подмножеств из исходных данных с возвращением) и случайного выбора признаков, что позволяет избежать переобучения и повысить обобщающую способность модели.

Случайный лес (Random Forest) — это мощный метод машинного обучения, который имеет свои преимущества и недостатки. Рассмотрим их подробнее.

Сильные стороны метода:

1. Высокая точность: Случайный лес часто демонстрирует высокую точность в задачах классификации и регрессии благодаря использованию ансамбля деревьев решений.
2. Устойчивость к переобучению: Благодаря бутстрэппингу и случайному выбору признаков, случайный лес менее подвержен переобучению по сравнению с одиночными деревьями решений.
3. Обработка больших объемов данных: Случайный лес может эффективно обрабатывать большие наборы данных с множеством признаков.
4. Интерпретируемость: Хотя интерпретировать ансамбль деревьев сложнее, чем одно дерево, случайный лес предоставляет важные метрики, такие как важность признаков, что помогает понять, какие факторы влияют на предсказания.
5. Работа с пропущенными значениями: Случайный лес может обрабатывать пропущенные значения в данных, что делает его более гибким.
6. Не требует строгих предпосылок: Случайный лес не требует строгих предпосылок о распределении данных, что делает его универсальным инструментом.

Слабые стороны метода:

1. Сложность модели: Случайный лес может быть сложным для интерпретации, особенно когда количество деревьев велико. Это может затруднить понимание, как модель принимает решения.
2. Время обучения: Обучение случайного леса может занять больше времени по сравнению с простыми моделями, особенно при большом количестве деревьев и признаков.
3. Потребление памяти: Случайный лес может потреблять значительное количество памяти, особенно при работе с большими наборами данных.
4. Сложности с предсказанием: В некоторых случаях, особенно при наличии сильной мультиколлинеарности, случайный лес может давать менее точные предсказания, чем более простые модели.
5. Сложности с балансировкой классов: Если классы сильно несбалансированы, случайный лес может смещаться в сторону более частого класса, что требует дополнительных методов балансировки.

**Градиентный бустинг**

Градиентный бустинг является одним из основных решений при работе с табличными, неоднородными данными, поскольку обладает высокой производительностью и точностью.

Этот мощный метод ансамблевого обучения используется для решения задач регрессии и классификации. Он строит модель, комбинируя несколько слабых моделей (обычно деревьев решений) для улучшения предсказаний.

Сильные стороны метода:

1. Высокая точность: Градиентный бустинг часто показывает отличные результаты на различных задачах, особенно в соревнованиях по машинному обучению.
2. Гибкость: Можно использовать различные функции потерь и модели базовых алгоритмов (например, деревья решений, линейные модели и т.д.).
3. Устойчивость к выбросам: Модели градиентного бустинга могут быть более устойчивыми к выбросам по сравнению с другими методами.

Слабые стороны метода:

1. Переобучение: Модели могут переобучаться, особенно если не использовать регуляризацию или если количество итераций слишком велико.
2. Время обучения: Обучение может занять много времени, особенно на больших наборах данных, так как каждая итерация требует обучения новой модели.
3. Сложность настройки: Градиентный бустинг имеет множество гиперпараметров, которые нужно настраивать, что может усложнить процесс обучения.

**Нейронные сети**

Нейронные сети — это вычислительные модели, вдохновленные биологическими нейронными сетями, которые используются для распознавания паттернов и решения задач, таких как классификация и регрессия. Основные компоненты нейронной сети включают:

- Нейроны: основные единицы обработки информации, которые принимают входные данные, применяют к ним весовые коэффициенты и функции активации, чтобы генерировать выходные данные;

- Слои: сети состоят из входного слоя, одного или нескольких скрытых слоев и выходного слоя. Каждый слой состоит из нейронов, которые обрабатывают данные и передают их следующему слою;

- Функции активации: функции (например, «ReLU», «Sigmoid») определяют, будет ли нейрон активирован (выдаст ли выходное значение) в зависимости от входных данных;

- Обучение: нейронные сети обучаются с использованием алгоритма обратного распространения ошибки («backpropagation»), который обновляет веса нейронов на основе разницы между предсказанными и фактическими значениями (ошибкой).

- Оптимизация: минимизации функции потерь используется оптимизатор (например, «Adam»), который корректирует веса нейронов в процессе обучения.

Сильные стороны метода:

1. Способность к обучению сложных паттернов: Нейронные сети могут моделировать сложные функции и выявлять нелинейные зависимости в данных.
2. Гибкость: Они могут быть адаптированы для различных задач, включая классификацию, регрессию, обработку изображений, текста и звука.
3. Автоматическое извлечение признаков: Нейронные сети могут автоматически извлекать важные признаки из необработанных данных, что уменьшает необходимость в ручной обработке данных.
4. Масштабируемость: Нейронные сети могут эффективно обрабатывать большие объемы данных и могут быть обучены на распределенных системах.

Слабые стороны метода:

1. Потребность в больших объемах данных: Для достижения хороших результатов нейронные сети требуют значительных объемов обучающих данных.
2. Переобучение: Нейронные сети могут легко переобучаться на небольших наборах данных, особенно если архитектура сети слишком сложна.
3. Долгое время обучения: Обучение нейронных сетей может занять много времени, особенно при использовании больших наборов данных и сложных архитектур.
4. Сложность настройки гиперпараметров: Нейронные сети имеют множество гиперпараметров (количество слоев, количество нейронов в каждом слое, скорость обучения и т.д.), которые требуют тщательной настройки.
5. Отсутствие интерпретируемости: Нейронные сети часто рассматриваются как "черные ящики", что затрудняет понимание того, как они принимают решения.

Нейронные сети являются мощным инструментом для решения различных задач в области машинного обучения, но их использование требует внимательного подхода к обучению и настройке.

**1.2.1 Условия работоспособности применяемых методов**

Условия, которые должны выполняться, чтобы определенный метод анализа или моделирования был применим и давал корректные результаты для различных методов машинного обучения, могут сильно различаться. Ниже, в Таблице 3, приведены основные априорные предпосылки к способности некоторых популярных методов.

Таблица 3 – Условия работоспособности методов

|  |  |
| --- | --- |
| **Метод** | **Условия** |
| Линейная регрессия | Зависимость между независимыми переменными и целью должна быть линейной. |
| Ошибки (остатки) должны быть нормально распределены |
| Вариация ошибок должна быть постоянной для всех уровней предсказанных значений |
| Ошибки должны быть независимы друг от друга |
| Не должно быть мультиколлинеарности: Независимые переменные не должны быть сильно скоррелированными |

Продолжение Таблицы 3

|  |  |
| --- | --- |
| **Метод** | **Условия** |
| Случайный лес | Наличие сильной корреляции между признаками может снизить эффективность модели |
| Работает лучше с большими объемами данных |
| Чувствителен к шуму в данных и может переобучаться на тренировочных данных. |
| Градиентный бустинг | Чувствителен к выбросам в данных, что может негативно сказаться на производительности |
| Сложность настройки модели |
| Требует значительных ресурсов памяти |
| Нейронные сети | Для обучения нейронных сетей требуется большое количество данных |
| Входные данные могут быть нелинейными, но функции активации могут требовать определенных свойств (например, непрерывности, линейности) |
| Данные должны быть нормализованы или стандартизированы для оптимальной работы. |

Каждый метод имеет свои собственные предпосылки. Очень важно перед применением метода провести анализ данных и проверить, выполняются ли эти предпосылки. Если предпосылки не выполняются, может потребоваться использовать другие методы или преобразовать данные, чтобы они соответствовали требованиям.

* 1. **Разведочный анализ данных (EDA)**

Анализ данных — это процесс систематического применения статистических и логических методов для описания, интерпретации и извлечения полезной информации из данных. Он включает в себя несколько ключевых этапов, каждый из которых играет важную роль в понимании и использовании данных. Анализ данных может требовать повторного обращения к предыдущим этапам для улучшения результатов и принятия более обоснованных решений.

Одним из этапов анализа данных, является разведывательный анализ данных (чаще используется сокращение «EDA» от англ. «exploratory data analysis»). Разведывательный анализ данных является критически важным этапом, который помогает исследователям и аналитикам лучше понять свои данные, выявить ключевые паттерны и подготовить почву для более сложных методов анализа, таких как моделирование и машинное обучение. Эффективное использование разведывательного анализа данных может значительно повысить качество и точность последующих аналитических шагов.

Основными целями «EDA» является понимание структуры данных (тип данных, распределений и статистических характеристик); выявление аномалий, выбросов и пропущенных значений; определение взаимосвязей (корреляций и зависимостей между переменными); формулирование гипотез на основе визуализации, а также подготовка данных для дальнейших этапов анализа.

* + 1. **Инструменты и методы (EDA)**

**Описание данных (метод df.info())**

Данный метод используется для получения информации о «DataFrame», в том числе: количество ненулевых значений в каждом столбце, тип данных каждого столбца, общее количество столбцов и строк, использование памяти.

Все эти методы и меры позволяет более глубоко понимать структуру и зависимости в данных, что, в свою очередь, помогает принимать более обоснованные аналитические решения.

**Визуализация данных**

Визуализация данных позволяет нам увидеть и понять паттерны, тренды и взаимосвязи в данных через графику и диаграммы. На таблице 3 приведены основные методы визуализации:

Таблица 3 - Визуализация данных

|  |  |
| --- | --- |
| Гистограмма | графическое представление распределения данных по различным интервалам. Она позволяет нам оценить, как часто значения попадают в определенные диапазоны и какие у нас имеются пики или провалы в данных |
| Диаграмма рассеяния | график, в котором каждая точка представляет собой отдельное наблюдение и показывает взаимосвязь между двумя переменными. Это помогает определить, есть ли какая-либо зависимость или корреляция между ними |
| Ящик с усами | статистические характеристики распределения данных, такие как медиана, квартили и выбросы. Он помогает быстро оценить разброс и симметрию данных, а также выявить потенциальные аномалии |
| Тепловая карта | графическое представление матрицы данных, где цветовая шкала показывает степень взаимосвязи между переменными |

Визуализация данных предоставляет нам мощные инструменты для исследования, анализа и визуализации информации из набора данных.

**Сводные статистики (метод «df.describe()»)**

Сводные статистики и меры центральной тенденции позволяют нам получить обобщенное представление о распределении данных и основных характеристиках. Это ключевые числовые метрики, которые помогают нам понять типичные и наиболее значимые значения в наборе данных. Основные параметры статистики представлены в таблице 4:

Таблица 4 – Основные параметры

|  |  |
| --- | --- |
| Среднее («Mean») | сумма всех значений, разделенная на их количество |
| Среднеквадратичное отклонение («Std») | показатель рассеивания значений случайной величины относительно её математического ожидания |
| Минимальное (min) значение | |
| Максимальное (maх)значение | |
| Медиана (Median) | серединное значение числового набора, упорядоченного по возрастанию. |
| Мода (Mode) | значение, которое встречается наиболее часто в наборе данных |
| 25%, 50%, 75% квантили | значение, которое заданная случайная величина не превышает с фиксированной вероятностью |

**Преобразование данных**

Преобразование данных – это процесс изменения шкалы или распределения переменных, чтобы сделать их более подходящими для анализа или моделирования. Это важный этап EDA, который помогает сгладить различия между переменными и создать более устойчивые и интерпретируемые данные. Одними из таких методов является нормализация и стандартизация.

Нормализация – это метод масштабирует значения переменных так, чтобы они находились в диапазоне от 0 до 1. Это особенно полезно, когда у нас есть переменные с разными единицами измерения и масштабами.

Стандартизация – метод преобразует значения переменных так, чтобы их среднее было равно 0, а стандартное отклонение – 1. Он делает распределение более "стандартным" и симметричным.

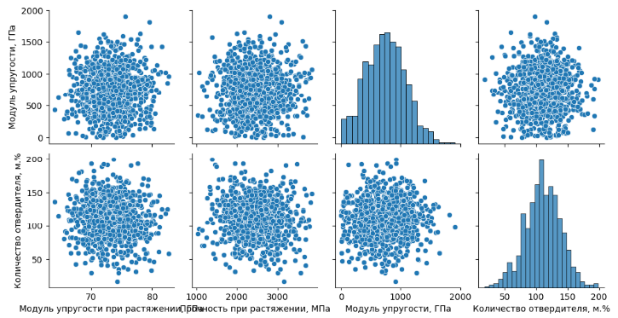
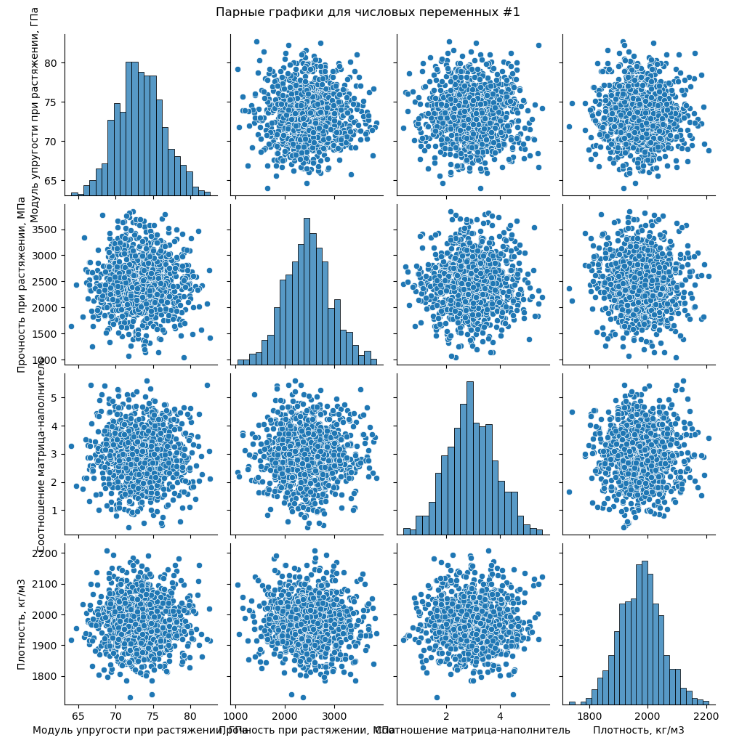
Преобразование данных может быть особенно полезным, когда у нас есть переменные с разными диапазонами значений, что может затруднять интерпретацию результатов. Это также может помочь алгоритмам машинного обучения работать более эффективно, так как они часто ожидают, что переменные будут иметь определенный масштаб или распределение.

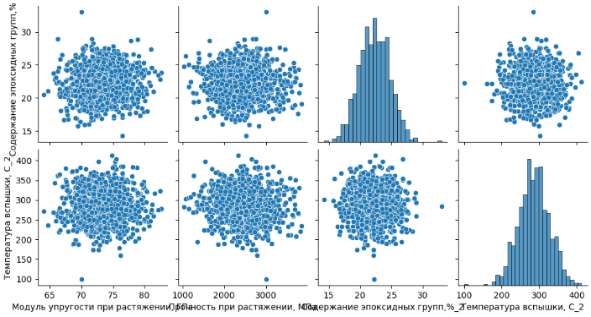
**2. Практическая часть**

**2.1 Предобработка данных**

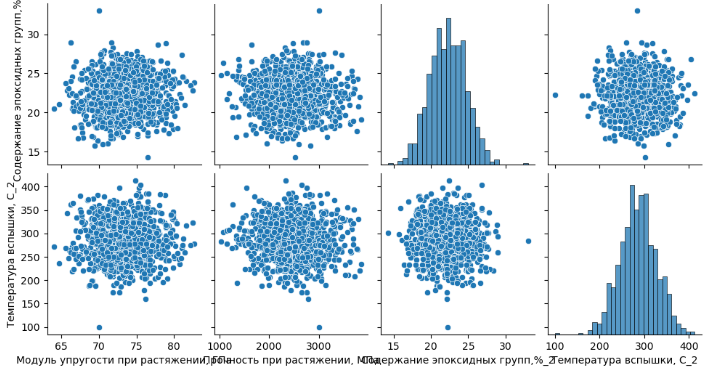
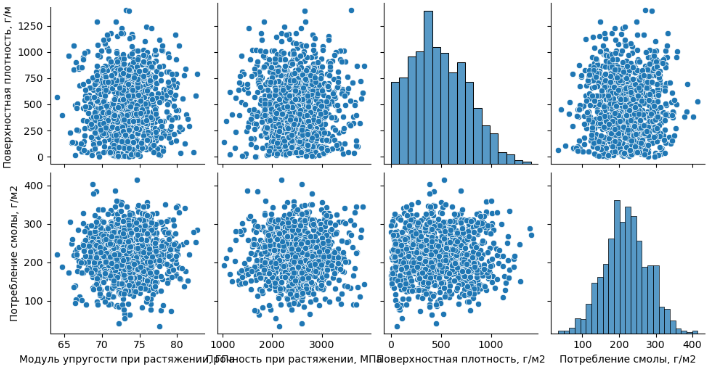
Ниже на рисунках приведены попарные графики для числовых переменных.

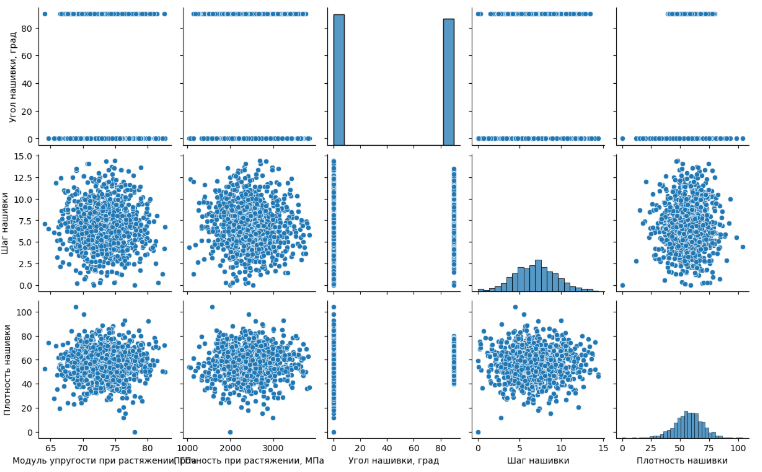
Графики 1





Продолжение Графиков 1





На основание выше приведенных графиков, делаем вывод, что данные распределены нормально, кроме столбца "Угол нашивки" - имеет всего два значения. Столбец с данными "Поверхностная плотность" имеет нормальное распределение со смещением влево.

Данные имеют облачный тип распределения – отсутствует корреляции (линейная зависимость) между признаками и таргетом. Отсутствует мультиколлинеарность признаков, что на данном этапе позволяет сохранить все признаки для дальнейшего анализа.

Облачное распределение данных говорит о сложной зависимости между признаками и таргетом, а также о том, что данные могут быть подвержены шуму.

Ниже на рисунках приведены попарные графики для числовых переменных после нормализации и таблице 5, с характеристикой данных, до и после нормализации:

Графики 2

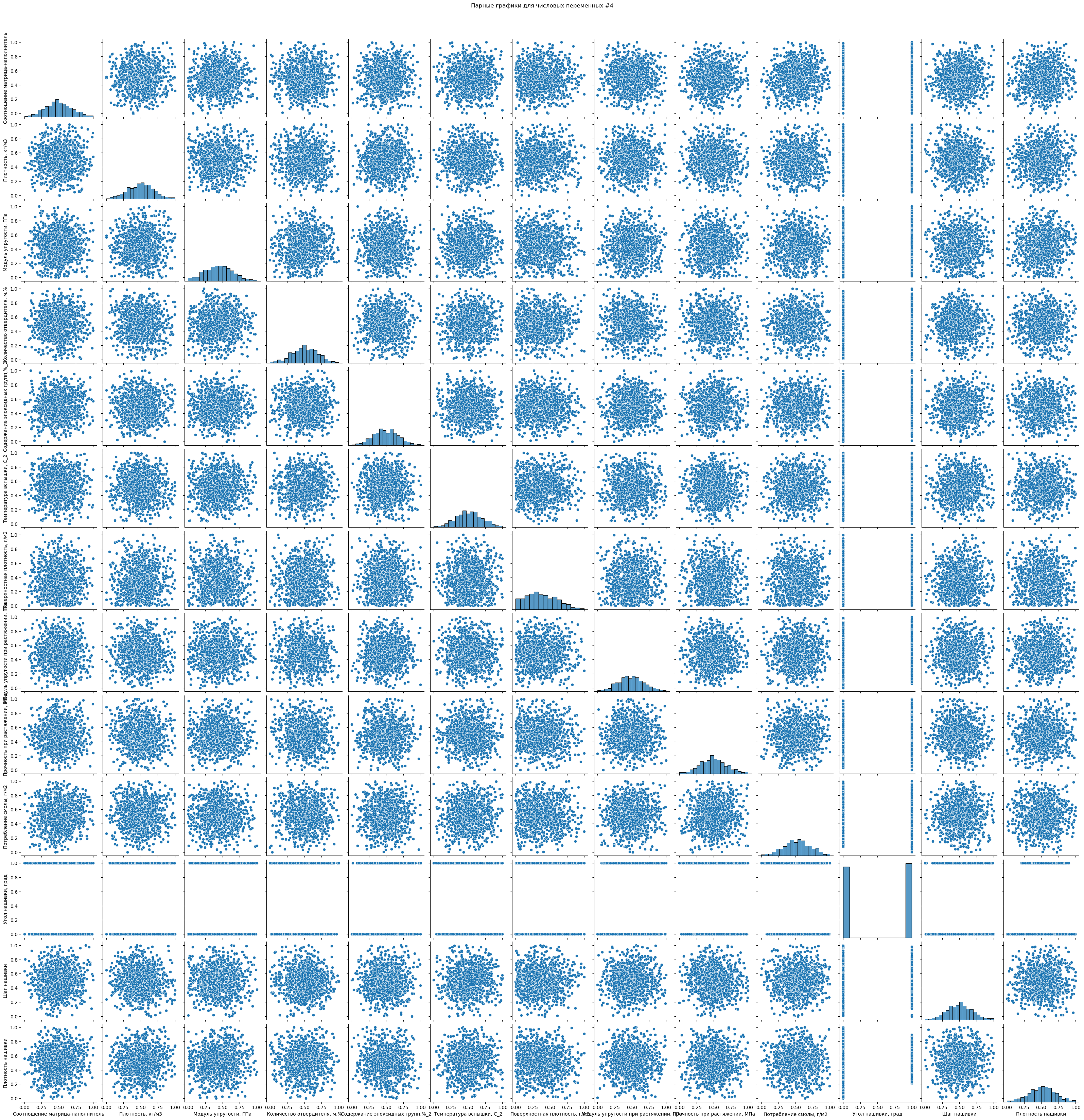


Таблица 5 – Минимальные и максимальные значения до и после нормализации

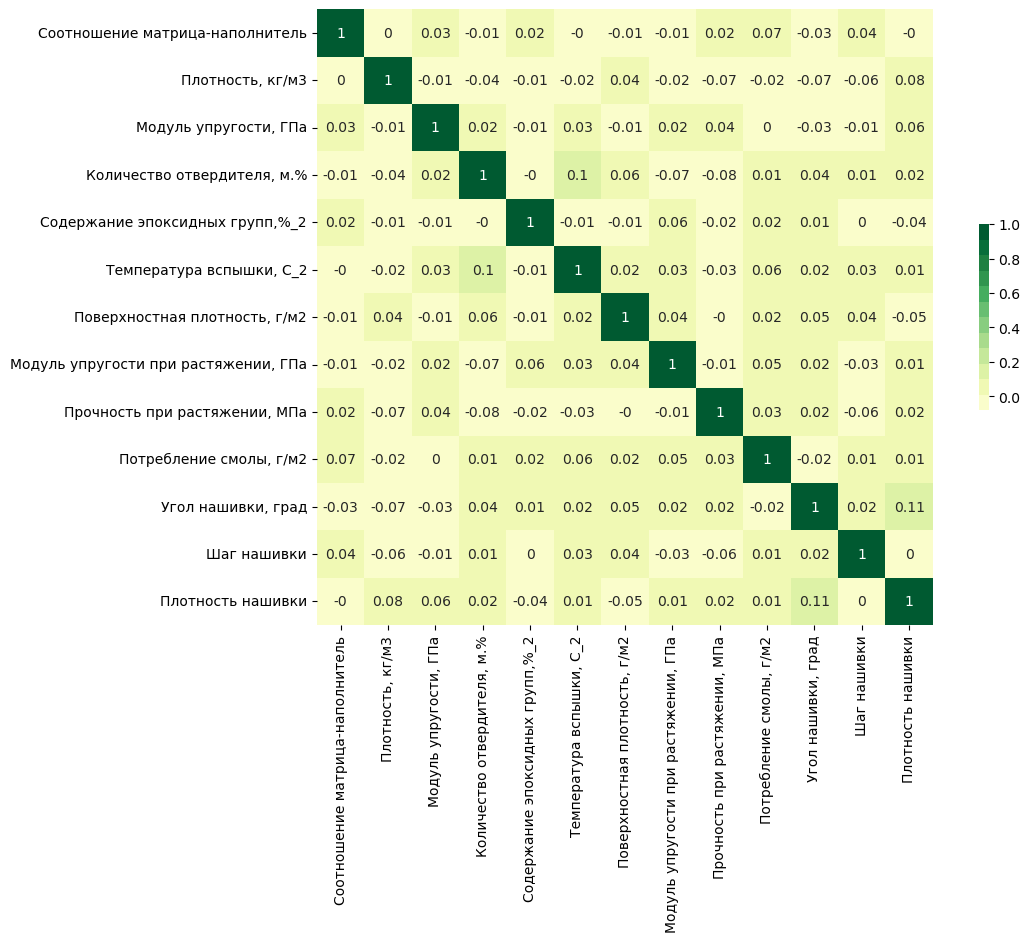
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Признаки | До | | После | |
| min | max | min | max |
| Соотношение матрица-наполнитель | 0.39 | 5.59 | 0 | 1 |
| Плотность, кг/м3 | 1731.76 | 2207.77 | 0 | 1 |
| Модуль упругости, ГПа | 2.44 | 1911.54 | 0 | 1 |

Продолжение Таблицы - 5

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Признаки | До | | После | |
| min | max | min | max |
| Количество отвердителя, м.% | 17.74 | 198.95 | 0 | 1 |
| Содержание эпоксидных групп, %\_2 | 14.25 | 33.00 | 0 | 1 |
| Температура вспышки, С | 100.00 | 413.27 | 0 | 1 |
| Поверхностная плотность, г/м2 | 0.60 | 1399.54 | 0 | 1 |
| Потребление смолы, г/м2 | 33.80 | 414.59 | 0 | 1 |
| Угол нашивки, град | 0.00 | 90.00 | 0 | 1 |
| Шаг нашивки | 0.00 | 14.44 | 0 | 1 |
| Плотность нашивки | 0.00 | 103.99 | 0 | 1 |

Для наглядности отсутствия линейной зависимости между нашими данными рассмотрим тепловую карта корреляции на рисунке ниже.

Рис.1 Тепловая карта



**2.1.1 Очистка данных от выбросов**

Для очистки данных использовали ящик с усами («Box Plot»). Это графический метод визуализации распределения данных, который позволяет быстро оценить основные статистические характеристики набора данных. Он показывает медиану, квартили и выбросы, что делает его полезным для анализа и сравнения распределений между несколькими группами.

Основная часть графика, которая представляет интерквартильный диапазон (IQR) — разницу между первым (Q1) и третьим (Q3) квартилями. Ящик показывает, где находится центральная половина данных.

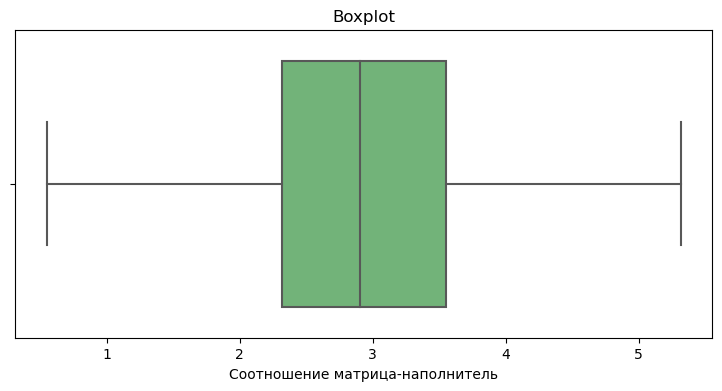
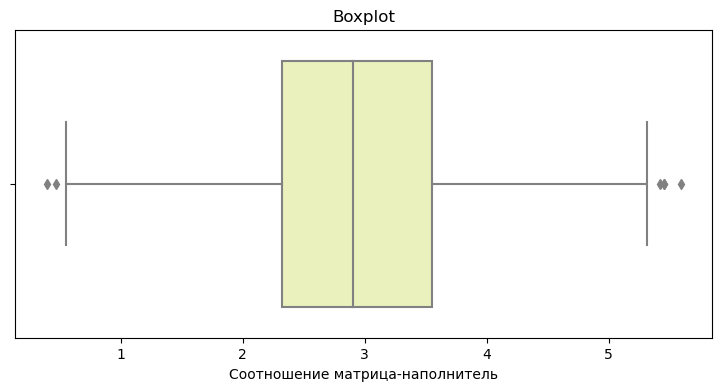
Медиана: линия внутри ящика, которая делит его на две части. Она представляет собой второй квартиль (Q2) и показывает среднее значение данных.

Усы: линии, которые выходят из ящика и показывают диапазон данных, не включая выбросы. Обычно усы простираются до 1.5 \* IQR от Q1 и Q3.

Выбросы: точки, которые находятся за пределами усов. Они могут указывать на аномалии или интересные особенности в данных.

Ниже на графиках можно видеть ящик с усами для одного из признаков до удаления и после удаления выбросов.

График 3



Все обнаруженные выбросы были удалены, так как их, по сравнению с общим количеством данных, немного и это не отразится негативно на данных.

Далее была проведена нормализация данных, о которой шла речь выше.

**2.2 Разработка и обучение модели**

Cложная, не линейная зависимость между признаками и таргетом в задаче регрессии предполагает использование сложных моделей. Классические регрессоры, при отсутствие линейной зависимости между таргетом и признаками, скорее всего не принесут ожидаемый результат.

Попробуем для входного эксперимента использовать следующие модели из библиотеки «sklearn»: «Градиентный бустинг» («GradientBoostingRegressor»), «Случайный лес» («RandomForestRegressor»), «Полиномиальная регрессия» («PolynomialFeatures»).

**2.3 Тестирование модели**

Для оценки качества моделей были выбраны три метрики: коэффициент детерминации («R2»), средняя абсолютная ошибка («MAE») и среднеквадратическая ошибка («MSE»).

Коэффициент детерминации («R2») – это единица минус доля необъяснённой дисперсии (дисперсии случайной ошибки модели, или условной по факторам дисперсии зависимой переменной) в дисперсии зависимой переменной. Его рассматривают как универсальную меру зависимости одной случайной величины от множества других.

Средняя абсолютная ошибка («MAE») – это степень несоответствия между фактическими и прогнозируемыми значениями. Она рассчитывается как среднее значение разности между спрогнозированным и фактическим значениями. Средняя абсолютная ошибка помогает понять эффективность модели.

Среднеквадратическая ошибка («MSE») – это функция потерь, которая используется в задачах регрессии. Она измеряет среднее квадратичное отклонение между истинными и предсказанными значениями. Она рассчитывается как разница между предсказанными значениями и истинными, возведенными в квадрат и усредненными по всему набору данных.

В таблице 6 представлены данные о метриках используемых моделей.

Таблица 6 Метрики моделей

|  |  |  |  |
| --- | --- | --- | --- |
| Модель | Метрики | | |
| «R2» | «MAE» | «MSE» |
| «GradientBoostingRegressor» | -0.08 | 2.45 | 9.45 |
| «RandomForestRegressor» | -0.03 | 2.43 | 8.95 |
| «PolynomialFeatures» | -0.08 | 2.47 | 9.39 |

Ниже на графиках 3 изображены предсказанные и фактические значения с линией идеальных предсказаний.

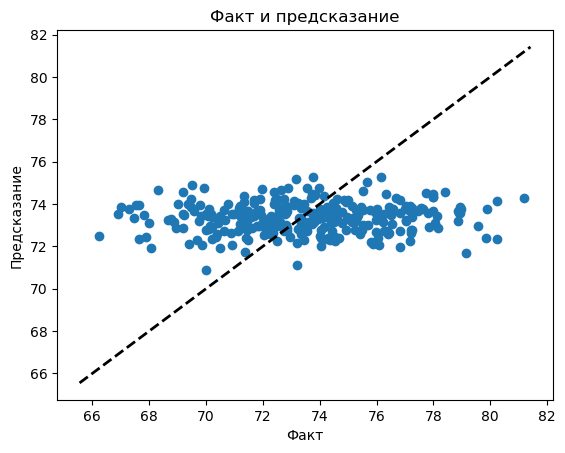
Графики 4

«GradientBoostingRegressor»

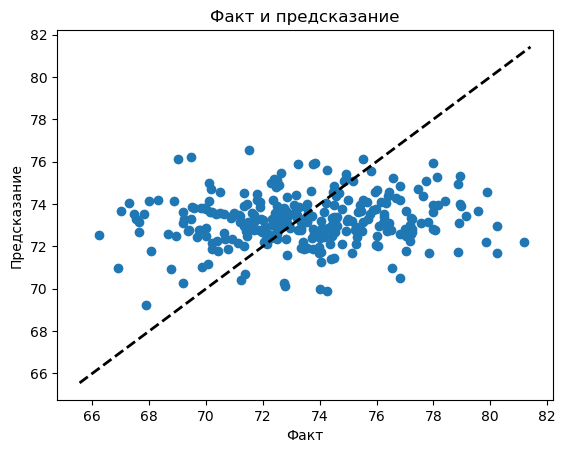


Продолжение Графика - 4

«RandomForestRegressor»



«PolynomialFeatures»



Коэффициент детерминации принимает в нашем случае отрицательное значение, это означает, что ошибка модели простого среднего становится меньше ошибки регрессионной модели.

В идеальном случае коэффициент детерминации должен стремиться к значению один, а ошибки к значению нуль.

В нашем случае модель не дает никого прогноза, что наглядно показывают графики, где мы видим, что данные распределены стохастически.

Разведывательный анализ предсказывал данный результат, данные имели облачное распределение и плохо коррелировали с таргетом. Нужно вернуться к начальному этапу и провести работу над данными или изменить постановку задачи.

**2.4 Смена постановки задачи**

Попробуем задачу регрессии превратить в задачу классификации с помощью задачи кластеризации.

Классификация – задача, в которой множество объектов необходимо разделить некоторым образом на классы, при этом задано конечное множество объектов, для которых известно, к каким классам они относятся (выборка), но классовая принадлежность остальных объектов неизвестна.

Кластеризация – задача, в которой задача группировки множества объектов на подмножества (кластеры) таким образом, чтобы объекты из одного кластера были более похожи друг на друга, чем на объекты из других кластеров по какому-либо критерию.

Разобьем наш таргет на кластеры, используя метод кластеризации К-средних («K-Means»). Каждый кластер будет включать интервал значений нашего таргета. После чего, используя кластеры как классы, попробуем провести классификацию наших данных.

Классификацию будем проводить с помощью методов: «К-ближайших соседей» («KNeighborsClassifier») и «Случайный лес» («RandomForestClassifier»).

**2.4.1. Описание используемых методов**

**К-средних (K-means)** — алгоритм кластеризации, который используется для разделения данных на K кластеров. Он работает на основе идеи, что объекты в одном кластере должны быть более похожи друг на друга, чем на объекты в других кластерах. Алгоритм минимизирует внутрикластерное расстояние, чтобы достичь компактных и хорошо разделенных кластеров.

Сильные стороны модели:

1. Простота и скорость: прост в реализации и работает быстро, особенно на больших наборах данных.
2. Эффективность: может эффективно обрабатывать большие объемы данных и хорошо масштабируется.
3. Гибкость: может быть адаптирован для различных типов данных и задач, а также позволяет легко изменять количество кластеров.

Слабые стороны модели:

1. Необходимость задания K: пользователь должен заранее определить количество кластеров (K), что может быть сложно без предварительного анализа данных.
2. Чувствительность к инициализации: результаты могут зависеть от начального выбора центроидов, что может привести к локальным минимумам.
3. Чувствительность к выбросам: может быть сильно подвержен влиянию выбросов, что может исказить результаты кластеризации.
4. Предположение о форме кластеров: предполагает, что кластеры имеют сферическую форму и равный размер, что не всегда соответствует реальным данным.

**К-ближайших соседей(«KNeighborsClassifieк»)**

Алгоритм используется для задач классификации и работает на основе идеи, что объекты, находящиеся близко друг к другу в пространстве признаков, имеют схожие классы. «KNeighborsClassifier» классифицирует новый объект на основе классов его k-ближайших соседей в обучающем наборе данных. Алгоритм вычисляет расстояние между объектом и всеми другими объектами в обучающем наборе, выбирает k ближайших соседей и определяет класс на основе большинства классов этих соседей.

Сильные стороны метода:

1. Простота и понятность: алгоритм легко понять и реализовать. Он не требует сложных математических вычислений.
2. Отсутствие предположений о распределении данных: не делает предположений о распределении данных, что делает его универсальным для различных типов данных.
3. Гибкость: можно использовать различные метрики расстояния (например, евклидово, манхэттенское и др.), что позволяет адаптировать алгоритм под конкретные задачи.
4. Хорошая производительность на малых и средних наборах данных: может показывать хорошие результаты на небольших и средних объемах данных.

Слабые стороны метода:

1. Высокая вычислительная сложность: при увеличении объема данных алгоритм становится медленным, так как для каждого нового объекта необходимо вычислять расстояния до всех обучающих примеров.
2. Проблемы с масштабированием: чувствителен к масштабированию признаков. Необходимо нормализовать или стандартизировать данные перед использованием алгоритма.
3. Проблемы с высокоразмерными данными: в высокоразмерных пространствах (проклятие размерности) расстояния между точками становятся менее информативными, что может ухудшить качество классификации.
4. Неустойчивость к шуму: может быть чувствителен к выбросам и шуму в данных, что может негативно сказаться на его производительности.

**Случайный лес («RandomForestClassifier»)**

Алгоритм машинного обучения, который относится к классу ансамблевых методов. Он создает множество деревьев решений, каждое из которых обучается на случайной подвыборке данных и случайном подмножестве признаков. При классификации нового объекта алгоритм агрегирует предсказания всех деревьев, обычно с помощью голосования, чтобы определить окончательный класс.

Сильные стороны метода:

1. Высокая точность: показывает высокую точность на различных задачах классификации благодаря использованию нескольких деревьев.
2. Устойчивость к переобучению: алгоритм менее подвержен переобучению по сравнению с одиночными деревьями решений, так как он усредняет результаты нескольких моделей.
3. Обработка больших данных: может эффективно обрабатывать большие объемы данных и высокоразмерные пространства.
4. Важность признаков: предоставляет информацию о важности признаков, что может помочь в интерпретации модели и выборе значимых переменных.
5. Отсутствие предположений о распределении данных: Random Forest не требует предположений о распределении данных, что делает его универсальным для различных типов данных.

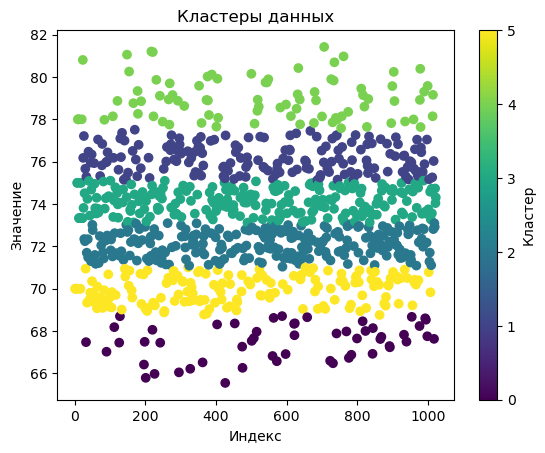
Слабые стороны метода:

1. Сложность интерпретации: интерпретировать предсказания модели может быть сложнее, чем в случае с одиночными деревьями решений.
2. Время обучения: обучение модели может занять больше времени по сравнению с одиночными деревьями, особенно при большом количестве деревьев.
3. Потребление памяти: может потреблять больше памяти, так как хранит множество деревьев решений.
4. Чувствительность к шуму: более устойчив к шуму, чем одиночные деревья, он все же может быть чувствителен к выбросам в данных.

**2.4.2. Разработка и тестирование модели**

Ниже представлена работа алгоритма K-средних, виде графика

Графики - 5



Как видно из графика 4, количество кластеров было выбрано шесть. В процессе использовались разные варианты разбивки, в том числе и ручным способом.

После кластеризации, которая для нас создала классы, приступаем к классификации.

Для оценки качества модели выводим матрицу ошибок и отчет классификации содержащий метрики точности («precision»), полноту («recall»), F-меру и «Accuracy».

Метрика «precision» (точность) также известна, как положительное прогнозное значение. Она измеряет вероятность того, что модель верно спрогнозировала, что значение является истинным. Это значение рассчитывается следующим образом: TP / (TP + FP), где TP — истинно-положительное решение; TN — истинно-отрицательное решение; FP — ложноположительное решение; FN — ложноотрицательное решение.

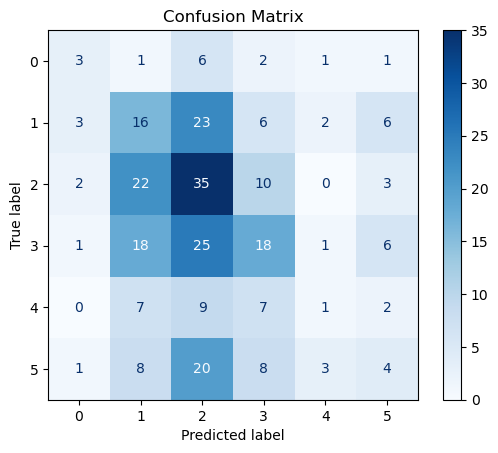
Метрика «Recall» (полнота) показывает, какую долю объектов положительного класса из всех объектов положительного класса нашёл алгоритм.

F-мера представляет собой гармоническое среднее между точностью и полнотой.

«Accuracy» — это показатель, который описывает общую точность предсказания модели по всем классам:

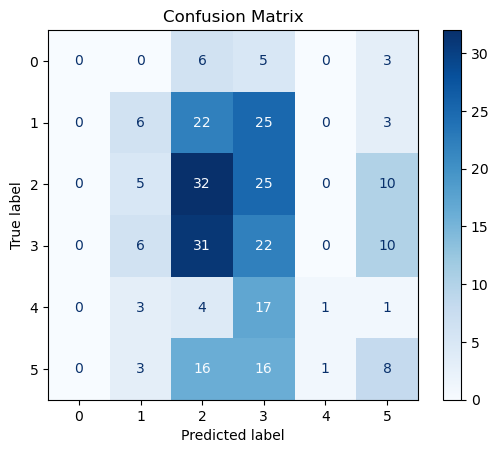
Отчет о классификации и матрица ошибок для метода

«KNeighborsClassifieк:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| класс | precision | recall | f1-score | accuracy |
| 0 | 0,00 | 0,00 | 0,00 | 0,20 |
| 1 | 0,16 | 0,14 | 0,15 |
| 2 | 0,19 | 0,24 | 0,21 |
| 3 | 0,22 | 0,28 | 0,25 |
| 4 | 0,20 | 0,04 | 0,06 |
| 5 | 0,24 | 0,23 | 0,23 |

Отчет о классификации и матрица ошибок для метода«RandomForestClassifier:



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| класс | precision | recall | f1-score | accuracy |
| 0 | 0,00 | 0,00 | 0,00 | 0,25 |
| 1 | 0,26 | 0,11 | 0,15 |
| 2 | 0,29 | 0,44 | 0,35 |
| 3 | 0,20 | 0,32 | 0,25 |
| 4 | 0,50 | 0,04 | 0,07 |
| 5 | 0,23 | 0,18 | 0,20 |

Отчет о классификации и матрица ошибок, к сожалению, говорят нам, что новый подход тоже не справился с задачей. Алгоритм не видит разницу в данных и зависимости межу ними, как и в предыдущих моделях.

**2.5 Нейронные сети**

В процессе выполнения работы мной был реализована нейронная сеть с использованием библиотеки «Keras» и «Sequential». Реализованные модели состоят из 3 слоев: два слоя Dense с 64 и 32 нейронами и функцией активации «relu» (нелинейная функция активации) и один выходной слой Dense с одним нейроном. Результаты метрик представлены в таблице 7

Таблице 7 Результаты метрик при прогнозировании параметра «Соотношение матрица-наполнитель»

|  |  |
| --- | --- |
| Коэффициент детерминации | -0,01 |
| Среднее абсолютное отклонение | 0,71 |
| Среднее квадратичное отклонение | 0,77 |

**2.6 Создание удаленного репозитория**

Репозиторий с проектом и всеми необходимыми файлами:

<https://github.com/ViMasHub/FQW_2025.git>

**Литература**

1. Грас Д. Data Science. Наука о данных с нуля /Пер. с англ. – 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2024 – 416с.: ил. –ISBN 978-5-9775-6731-2.
2. Основы Python. Научитесь думать, как программист / Аллен Б. Дауни; пер. с англ. С. Черникова; [науч. Ред. А. Родионов]. –Москва:Манн, Иванов и Фербер, 2023 – 304 с. –(O’Reilly). –ISBN 978-5-00146-798-
3. Ляпков, А. А. Полимерные аддитивные технологии / А. А. Ляпков, А. А. Троян. — 2-е изд., стер. — Санкт-Петербург: Лань, 2024. — 120 с. — ISBN 978-5-507-48773-8.