ЦЕНТР ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ МГТУ им. Н.Э. Баумана

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

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

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

# Прогнозирование конечных свойств новых материалов (композиционных материалов)

Слушатель Кирпичев Артем Игоревич

Москва, 2025

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

# Содержание

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

**Тема:**

Прогнозирование конечных свойств новых материалов (композиционных материалов).

**Описание:**

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).

На входе имеются данные о начальных свойствах компонентов композиционных материалов (количество связующего, наполнителя,

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

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).

Был проведен разведочный анализ предложенных данных, нарисованы гистограммы распределения каждой из переменной, диаграммы ящика с усами, попарные графики рассеяния точек, получены для каждой колонки среднее, медианное значение, проведен анализ и исключены выбросы, была проведена проверка на наличие пропусков.

В ходе проведения исследований было создано модели, позволяющих предсказывать показатели упругости и прочности материалов при растяжении. Были разработаны две нейронные сети для прогноза модуля упругости при растяжении и прочности при растяжении.  Так же было создано веб-приложение на фреймворке Flask.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

# 1. Аналитическая часть

## 1.1. Постановка задачи

В ходе работы были изучены два предоставленных файла: X_bp.xlsx (с данными о параметрах базальтопластика, состоящий из 1024 строки и 11 столбцов) и X_nup.xlsx (данные нашивок углепластика, состоящий из 1041 строки и 4 столбцов).

| | Соотношение матрица-наполнитель | Плотность, кг/м3 | модуль упругости, ГПа | Количество отвердителя, м.% | Содержание эпоксидных групп,%_2 | Температура вспышки, С_2 | Поверхностная плотность, г/м2 | Модуль упругости при растяжении, ГПа | Прочность при растяжении, МПа | Потребление смолы, г/м2 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1.857143 | 2030.0 | 738.736842 | 30.00 | 22.267857 | 100.000000 | 210.0 | 70.0 | 3000.0 | 220.0 |
| 1 | 1.857143 | 2030.0 | 738.736842 | 50.00 | 23.750000 | 284.615385 | 210.0 | 70.0 | 3000.0 | 220.0 |
| 2 | 1.857143 | 2030.0 | 738.736842 | 49.90 | 33.000000 | 284.615385 | 210.0 | 70.0 | 3000.0 | 220.0 |
| 3 | 1.857143 | 2030.0 | 738.736842 | 129.00 | 21.250000 | 300.000000 | 210.0 | 70.0 | 3000.0 | 220.0 |
| 4 | 2.771331 | 2030.0 | 753.000000 | 111.86 | 22.267857 | 284.615385 | 210.0 | 70.0 | 3000.0 | 220.0 |

Рисунок 1 - датасет X_bp.xlsx

| | Угол нашивки, град | Шаг нашивки | Плотность нашивки |
|---|---|---|---|
| 0 | 0.0 | 4.0 | 57.0 |
| 1 | 0.0 | 4.0 | 60.0 |
| 2 | 0.0 | 4.0 | 70.0 |
| 3 | 0.0 | 5.0 | 47.0 |
| 4 | 0.0 | 5.0 | 57.0 |

Рисунок 2 - датасет X_nup.xlsx

Задача разработать модель, чтобы спрогнозировать модуль упругости при растяжении, прочности при растяжении и соотношения матрица-наполнитель. Делаем объединение датасетов по индексу, тип объединения INNER.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

| | Соотношение матрица-наполнитель | Плотность, кг/м3 | модуль упругости, ГПа | Количество отвердителя, м.% | Содержание эпоксидных групп,%_2 | Температура вспышки, С_2 | Поверхностная плотность, г/м2 | Модуль упругости при растяжении, ГПа | Прочность при растяжении, МПа | Потребление смолы, г/м2 | Угол нашивки, град | Шаг нашивки | Плотность нашивки |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1.857143 | 2030.000000 | 738.736842 | 30.000000 | 22.267857 | 100.000000 | 210.000000 | 70.000000 | 3000.000000 | 220.000000 | 0.0 | 4.000000 | 57.000000 |
| 1 | 1.857143 | 2030.000000 | 738.736842 | 50.000000 | 23.750000 | 284.615385 | 210.000000 | 70.000000 | 3000.000000 | 220.000000 | 0.0 | 4.000000 | 60.000000 |
| 2 | 1.857143 | 2030.000000 | 738.736842 | 49.900000 | 33.000000 | 284.615385 | 210.000000 | 70.000000 | 3000.000000 | 220.000000 | 0.0 | 4.000000 | 70.000000 |
| 3 | 1.857143 | 2030.000000 | 738.736842 | 129.000000 | 21.250000 | 300.000000 | 210.000000 | 70.000000 | 3000.000000 | 220.000000 | 0.0 | 5.000000 | 47.000000 |
| 4 | 2.771331 | 2030.000000 | 753.000000 | 111.860000 | 22.267857 | 284.615385 | 210.000000 | 70.000000 | 3000.000000 | 220.000000 | 0.0 | 5.000000 | 57.000000 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 1018 | 2.271346 | 1952.087902 | 912.855545 | 86.992183 | 20.123249 | 324.774576 | 209.198700 | 73.090961 | 2387.292495 | 125.007669 | 90.0 | 9.076380 | 47.019770 |
| 1019 | 3.444022 | 2050.089171 | 444.732634 | 145.981978 | 19.599769 | 254.215401 | 350.660830 | 72.920827 | 2360.392784 | 117.730099 | 90.0 | 10.565614 | 53.750790 |
| 1020 | 3.280604 | 1972.372865 | 416.836524 | 110.533477 | 23.957502 | 248.423047 | 740.142791 | 74.734344 | 2662.906040 | 236.606764 | 90.0 | 4.161154 | 67.629684 |
| 1021 | 3.705351 | 2066.799773 | 741.475517 | 141.397963 | 19.246945 | 275.779840 | 641.468152 | 74.042708 | 2071.715856 | 197.126067 | 90.0 | 6.313201 | 58.261074 |
| 1022 | 3.808020 | 1890.413468 | 417.316232 | 129.183416 | 27.474763 | 300.952708 | 758.747882 | 74.309704 | 2856.328932 | 194.754342 | 90.0 | 6.078902 | 77.434468 |

Рисунок 3 – объединение таблиц

Проводим разведочный анализ данных, рисуем гистограммы распределения для все переменных, графики рассеивания и диаграммы размаха(boxplot), проверка наличия пропусков, уникальных значений в колонках, удаления шумов и выбросов, а также нормализация и стандартизация.

Обучаем модель для прогноза модуля упругости при растяжении и прочности при растяжении. Создаем нейронную сеть, которая будет рекомендовать соотношение матрица-наполнитель. Оцениваем точность модели. Создаем приложение на фреймворке Flask.

Весь код работы размещаем в GitHib.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

## 1.2 Описание используемых методов

Данная задача относится к категории машинного обучения с учителем и представляет собой классическую задачу регрессии. В рамках регрессионного анализа основной целью любого алгоритма является определение функции потерь и её минимизация для достижения оптимального результата. Такой подход позволил достичь оптимального баланса между точностью предсказаний и обобщающей способностью модели.

В процессе исследования для решения поставленной задачи были применены следующие методы:

1. К-ближайших соседей;
2. дерево решений;
3. стохастический градиентный спуск;
4. многослойный перцептрон;
5. Лассо;
6. метод опорных векторов;
7. случайный лес;
8. линейная регрессия;
9. градиентный бустинг;

Метод опорных векторов (Support Vector Regression, SVR) был выбран для решения задачи, поскольку демонстрирует высокую эффективность при работе с небольшими наборами данных.

Данный алгоритм относится к классу контролируемого машинного обучения и применяется как для задач классификации, так и для регрессионного анализа. Принцип работы метода основан на использовании схожих алгоритмов для анализа данных и распознавания паттернов.

Основной принцип работы заключается в следующем: на основе обучающей выборки алгоритм присваивает каждому объекту метку принадлежности к одной из двух категорий. Затем строится математическая модель, позволяющая классифицировать новые наблюдения.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

Математическая основа метода опорных векторов представляет собой отображение данных в многомерное пространство таким образом, что между точками разных категорий формируется максимальный разделительный зазор. Это позволяет:

- Эффективно разделять классы данных
- Минимизировать вероятность ошибок классификации
- Обеспечивать хорошую обобщающую способность модели

Такой подход обеспечивает высокую точность классификации при относительно небольших объёмах, обучающих данных.

### Математическое представление данных

В методе опорных векторов каждый объект данных представляется в виде вектора (точки) в p-мерном пространстве. На основе этих точек алгоритм строит разделяющую гиперплоскость (в двумерном случае - линию), которая оптимально разделяет данные на различные классы.

### Преимущества метода

- Эффективность обучения: метод требует относительно небольшого набора обучающих данных для построения качественной модели
- Универсальность применения: возможность успешного использования на реальных данных при корректной настройке на тестовом множестве
- Гибкость настройки: возможность тонкой настройки разделяющей функции под конкретную задачу
- Устойчивость к размерности: эффективность работы при большом количестве гиперпараметров
- Уникальная способность: возможность обработки случаев, когда количество гиперпараметров превышает число наблюдений
- Оптимизация классификации: алгоритм максимизирует ширину разделяющей полосы, что минимизирует вероятность ошибок классификации (аналогично работе подушки безопасности)

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

### Ограничения метода

- Чувствительность к шуму: высокая восприимчивость к выбросам в данных, что потребовало проведения тщательной предобработки и очистки данных от аномалий
- Временные затраты: значительное время обучения при работе с большими наборами данных
- Сложность трансформации: трудности при подборе эффективных преобразований данных
- Проблемы интерпретируемости: сложность интерпретации параметров модели

Из-за указанных ограничений в рамках исследования были рассмотрены и другие методы машинного обучения для сравнения их эффективности, и применимости к конкретной задаче.


Рисунок 4 - график метода опорных векторов для прочности при растяжении, Мпа

### Случайный лес (Random Forest)

Random Forest представляет собой ансамблевый метод машинного обучения с учителем, основанный на совокупности решающих деревьев. Данный алгоритм является универсальным решением для задач классификации и регрессии, особенно эффективным в случаях, когда одиночное дерево решений демонстрирует недостаточную точность.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

## Принцип работы

Метод строится на принципе объединения множества отдельных моделей в единый ансамбль, что позволяет значительно улучшить качество предсказаний по сравнению с использованием одиночного дерева решений.

## Преимущества метода

- Устойчивость к переобучению: модель практически не подвержена переобучению
- Минимальные требования к данным: отсутствие необходимости в предобработке входных данных
- Универсальность применения:
  - Эффективная обработка пропущенных значений
  - Работа с большим количеством классов и признаков
- Высокая производительность:
  - Точность предсказаний
  - Встроенная оценка обобщающей способности
  - Хорошая масштабируемость
  - Возможность параллельных вычислений

## Ограничения метода

- Вычислительная сложность: значительные временные затраты на построение модели
- Сложность интерпретации: трудности с интерпретацией результатов
- Ограничения применения:
  - Отсутствие возможности экстраполяции
  - Риск недостаточного обучения модели
  - Сложность прогнозирования результатов
- Сравнительная эффективность: в некоторых случаях может уступать линейным методам

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

Несмотря на определённые ограничения, метод случайного леса остаётся одним из наиболее популярных и эффективных инструментов машинного обучения благодаря своей универсальности и высокой точности предсказаний.

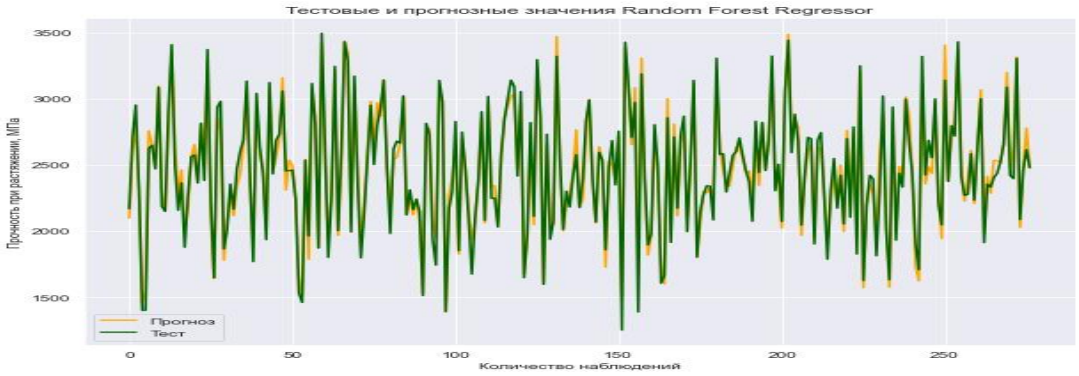


Рисунок 5 - график "случайного леса" для прочности при растяжении, Мпа

## Линейная регрессия (Linear Regression)

Линейная регрессия представляет собой базовый алгоритм контролируемого машинного обучения, который анализирует взаимосвязь между одной или несколькими входными переменными и выходной переменной. Это один из наиболее простых и эффективных инструментов статистического моделирования.

## Принцип работы

Метод основан на построении линии наилучшего соответствия, которая описывает зависимость между переменными. Модель генерирует несколько ключевых метрик для оценки качества:

- Коэффициент детерминации (R2) — показатель, демонстрирующий способность модели объяснять дисперсию данных
- Интерпретация значений R2:
  - R2=1 — модель идеально описывает все данные
  - R2=0,5 — модель объясняет 50% дисперсии данных
  - Чем ближе значение к 1, тем лучше качество модели

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

Преимущества метода

- Простота реализации: быстрая и лёгкая в программировании модель
- Интерпретируемость: результаты легко понять и объяснить
- Вычислительная эффективность: меньшая сложность по сравнению с другими алгоритмами
- Универсальность: подходит для базового анализа данных

Ограничения метода

- Линейность зависимостей: возможность моделирования только прямых линейных связей
- Требования к данным: необходимость наличия прямой корреляции между зависимыми и независимыми переменными
- Чувствительность к выбросам: значительное влияние аномальных значений на результаты
- Ограниченность границ: линейные ограничения прогнозируемых значений

Несмотря на ограничения, линейная регрессия остаётся важным инструментом в арсенале аналитика данных, особенно на начальных этапах исследования и при работе с относительно простыми зависимостями.

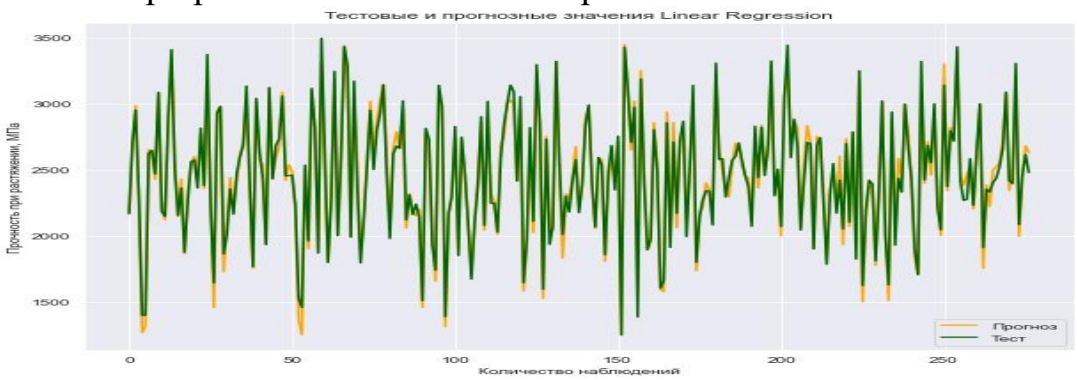


Рисунок 6 - график линейной регрессии для прочности при растяжении, Мпа

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

## Градиентный бустинг (Gradient Boosting)

Градиентный бустинг представляет собой мощный ансамблевый метод машинного обучения, основанный на последовательном построении ансамбля деревьев решений. Алгоритм использует технику градиентного спуска для минимизации функции потерь.

### Принцип работы

Метод основан на последовательном обучении базовых классификаторов, где каждый последующий классификатор стремится компенсировать ошибки предыдущих. Итоговая модель формируется как линейная комбинация всех базовых классификаторов.

### Ключевые особенности алгоритма

- Итеративное обучение: последовательное построение деревьев решений
- Компенсация ошибок: каждый новый классификатор корректирует недостатки предыдущих
- Минимизация потерь: оптимизация через градиентный спуск

### Преимущества метода

- Адаптивное обучение: новые алгоритмы обучаются на ошибках предыдущих моделей
- Эффективность оптимизации: требует меньшего количества итераций для достижения точных прогнозов
- Целевая выборка: наблюдения выбираются на основе величины ошибки
- Простота настройки: лёгкая настройка темпа обучения и параметров
- Интерпретируемость: результаты поддаются логическому объяснению

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

### Ограничения метода

- Риск переобучения: необходимость тщательного выбора критериев остановки обучения
- Дисбаланс данных: наблюдения с большими ошибками могут получать избыточное внимание
- Ограниченная гибкость: уступает нейронным сетям в сложности решаемых задач
- Чувствительность к гиперпараметрам: требует внимательной настройки параметров

Градиентный бустинг остаётся одним из наиболее эффективных алгоритмов машинного обучения, особенно в задачах, где требуется высокая точность предсказаний при сохранении разумной интерпретируемости модели.


Рисунок 7 - график градиентного бустинга для прочности при растяжении, МПа

### Метод k-ближайших соседей (kNN)

K-ближайших соседей (k Nearest Neighbours) — это непараметрический метод машинного обучения, основанный на принципе близости объектов в пространстве признаков. Алгоритм хранит весь набор обучающих данных и использует их для классификации или регрессии новых объектов.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

## Принцип работы

Алгоритм работает следующим образом:

- Вычисляет расстояния между новым объектом и всеми объектами в обучающей выборке
- Отбирает k ближайших соседей
- Принимает решение на основе:
  - большинства голосов (для классификации)
  - усреднения значений (для регрессии)

## Преимущества метода

- Простота реализации: легко понять и внедрить
- Интерпретируемость: результаты легко объяснимы
- Устойчивость к выбросам: низкая чувствительность к аномальным значениям
- Минимальные требования: не требует построения сложной модели
- Гибкость настройки: возможность настройки ключевых параметров
- Универсальность применения: подходит для различных типов задач
- Эффективность: находит оптимальное решение в рамках своей парадигмы
- Масштабируемость: хорошо работает с задачами небольшой размерности

## Ограничения метода

- Вычислительная сложность: значительное замедление при росте объёма данных
- Отсутствие обобщения: не формирует явных правил классификации
- Зависимость от данных: полностью опирается на весь массив исторических данных

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

- Непрозрачность решений: сложность объяснения оснований принятых решений
- Сложность метрики: трудности с выбором оптимальной метрики расстояния
- Высокая чувствительность: зависимость результатов от выбранной метрики
- Полные перебор: необходимость перебора всей обучающей выборки при классификации
- Вычислительная нагрузка: высокая ресурсоёмкость при работе с большими наборами данных

Несмотря на свои ограничения, метод k-ближайших соседей остаётся полезным инструментом в арсенале аналитика данных, особенно при работе с небольшими наборами данных и когда требуется простое, но эффективное решение задачи классификации или регрессии.


Рисунок 8 - график К-ближайших соседей для прочности при растяжении, Мпа

## Дерево решений (DecisionTreeRegressor)

Дерево решений представляет собой мощный инструмент контролируемого машинного обучения, предназначенный для автоматического анализа больших массивов данных. Метод использует древовидную структуру, напоминающую блок-схему, для моделирования процесса принятия решений и оценки их возможных результатов.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

## Принцип работы

Древовидная структура позволяет:

- Прогнозировать результаты на основе входных данных
- Оценивать затраты и полезность различных решений
- Обрабатывать как непрерывные, так и категориальные выходные переменные

Алгоритм генерирует правила на основе обобщения обучающих примеров и обучается прогнозировать будущие значения путём отслеживания характеристик объектов.

## Особенности применения

- Интеллектуальный анализ: эффективный инструмент для работы с данными
- Предсказательная аналитика: способность прогнозировать результаты
- Гибкость применения: подходит для регрессионных задач без явной корреляции
- Генерация правил: создание чётких правил, на основе обучающих данных

## Преимущества метода

- Визуализация процесса: наглядное представление принятия решений
- Последовательность решений: учёт влияния предыдущих решений на последующие
- Простота использования:
- Понятные правила работы
- Лёгкость интерпретации результатов
- Автоматическая обработка пропущенных значений
- Универсальность:
- Работа с различными типами переменных

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

- Выделение ключевых факторов прогнозирования

Ограничения метода

- Точность классификации: возможные ошибки при большом количестве классов и малой выборке
- Стабильность модели:
- Высокая чувствительность к изменениям
- Возможность построения существенно разных деревьев при небольших изменениях
- Вычислительная сложность:
- Затратные вычисления
- Необходимость контроля размера дерева
- Ограниченность решений: ограниченное число вариантов решения проблемы

Дерево решений остаётся важным инструментом в машинном обучении благодаря своей наглядности и способности решать сложные задачи прогнозирования, несмотря на определённые ограничения в применении.

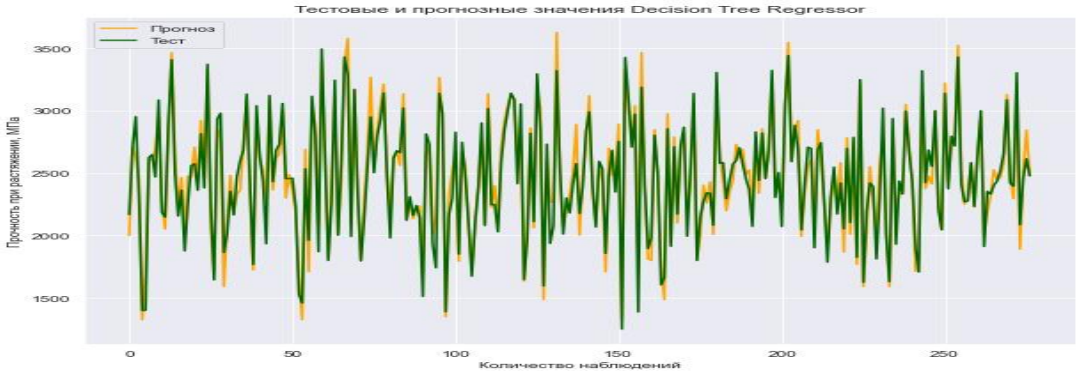


Рисунок 1 - график дерева принятия решений для прочности при растяжении

Стохастический градиентный спуск (SGDRegressor)

Стохастический градиентный спуск (Stochastic Gradient Descent, SGD) — это эффективный алгоритм оптимизации, используемый для обучения линейных классификаторов и регрессоров при работе с выпуклыми функциями потерь. Метод основан на корректировке весов модели с использованием

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

приближённого градиента, вычисленного на основе одного случайного обучающего примера.

### Принцип работы

Алгоритм функционирует следующим образом:

- Выбирает случайный пример из обучающей выборки
- Вычисляет градиент функции потерь на основе этого примера
- Корректирует веса модели в направлении, противоположном градиенту
- Повторяет процесс для каждого примера в выборке

### Преимущества метода

- Эффективность: быстрое обучение моделей
- Простота реализации: лёгкость внедрения алгоритма
- Гибкость настройки: множество параметров для оптимизации кода
- Масштабируемость: способность работать с очень большими наборами данных
- Универсальность применения: подходит для различных задач машинного обучения

### Ограничения метода

- Зависимость от гиперпараметров:
- Требует тщательной настройки параметров
- Чувствительность к масштабированию признаков
- Проблемы сходимости:
- Возможность медленной сходимости
- Риск застревания в локальных минимумах
- Вероятность отсутствия сходимости
- Риск переобучения: возможность чрезмерной подгонки под обучающие данные

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

- Сложность оптимизации: многоэкстремальность функционала

### Практическое применение

Несмотря на определённые ограничения, стохастический градиентный спуск остаётся одним из наиболее популярных методов оптимизации в машинном обучении благодаря своей эффективности и способности работать с большими объёмами данных. При правильной настройке параметров и предварительной обработке данных метод позволяет достигать высоких результатов в решении различных задач прогнозирования и классификации.


Рисунок 2-график стохастического градиентного спуска для модуля упругости

### Многослойный персептрон (MLPRegressor)

Многослойный персептрон представляет собой алгоритм контролируемого машинного обучения, который изучает функцию $f(\cdot):Rm \rightarrow Ro$, где:

- m — количество входных измерений
- o — количество выходных измерений

### Архитектура сети

Нейронная сеть состоит из:

- Входного слоя — принимает исходные данные
- Скрытых слоёв — один или более слоёв для обработки данных
- Выходного слоя — формирует итоговый результат

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

## Принцип работы

Алгоритм функционирует на основе:

- Обратного распространения ошибки
- Корректировки весов соединений
- Использования функций активации
- Минимизации функции потерь

## Преимущества метода

- Гибкость модели: способность строить сложные разделяющие поверхности
- Универсальность: возможность реализации практически любого отображения входных векторов в выходные
- Обобщающая способность: эффективное обобщение на новые данные
- Нелинейность: способность изучать сложные нелинейные зависимости
- Независимость: отсутствие требований к распределению входных данных

## Ограничения метода

- Сложность оптимизации: наличие невыпуклой функции потерь
- Чувствительность к инициализации: различные начальные веса могут давать разные результаты
- Настройка параметров: необходимость тщательной настройки гиперпараметров
- Масштабирование: чувствительность к масштабированию входных признаков
- Локальные минимумы: риск застревания в локальных минимумах при оптимизации

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

### Практическое применение

Многослойный персептрон является мощным инструментом для решения сложных задач машинного обучения, особенно когда требуется моделирование нелинейных зависимостей. При правильной настройке и предварительной обработке данных метод позволяет достигать высоких результатов в различных областях применения.

Для достижения оптимальных результатов рекомендуется:

- Проводить масштабирование входных данных
- Тщательно подбирать гиперпараметры
- Использовать методы регуляризации
- Контролировать процесс обучения

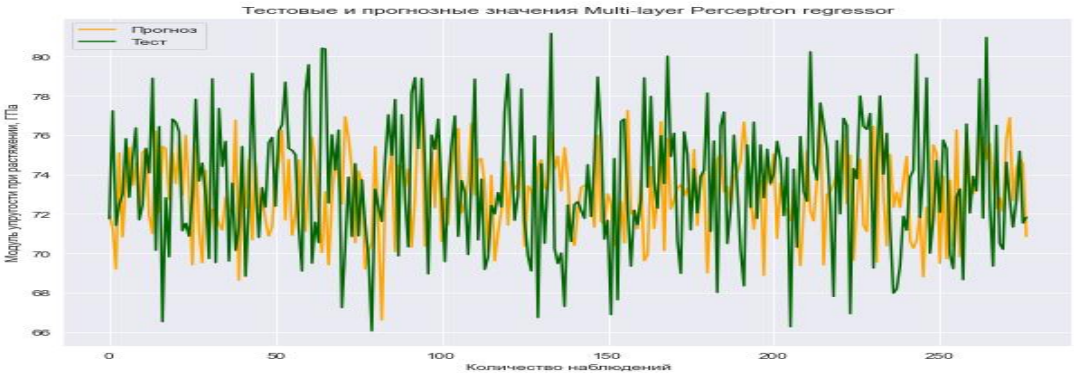


Рисунок 11 - график многослойного персептрона для модуля упругости, Гпа

### Лассо регрессия (Lasso)

Лассо регрессия — это линейная модель машинного обучения, предназначенная для оценки разреженных коэффициентов. Метод позволяет:

- Уменьшить сложность модели
- Предотвратить переобучение
- Получить более устойчивое решение

### Принцип работы

Метод основан на введении L1-регуляризации в функцию оптимизации, что позволяет:

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

- Снизить коллинеарность между признаками
- Уменьшить дисперсию модели
- Автоматически проводить отбор признаков

В отличие от классической линейной регрессии, Лассо использует абсолютное значение коэффициентов вместо квадратичного смещения.

### Применение

Особенно эффективно Лассо регрессия показывает себя в задачах:

- Прогнозирования временных рядов
- Авторегрессионных моделей
- Отбора значимых признаков

### Преимущества метода

- Фильтрация шумов: эффективное удаление шума из данных
- Вычислительная эффективность: быстрая работа алгоритма
- Низкая ресурсоёмкость: минимальные требования к вычислительным ресурсам
- Автоматизация отбора: способность исключать нерелевантные признаки
- Простота интерпретации: возможность обнуления коэффициентов

### Ограничения метода

- Выбор модели: может негативно влиять на качество модели
- Точность прогнозов: возможное снижение качества предсказаний
- Ложные срабатывания: риск получения некорректных результатов
- Коллинеарность: случайный выбор среди коррелированных переменных
- Форма взаимосвязи: не оценивает корректность связи между переменными

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

- Сравнительная эффективность: не всегда превосходит пошаговую регрессию

### Практические рекомендации

Для эффективного применения Лассо регрессии рекомендуется:

- Тщательно подбирать параметр регуляризации
- Проводить предварительную стандартизацию данных
- Оценивать качество модели на валидационной выборке
- Сравнивать результаты с другими методами регуляризации

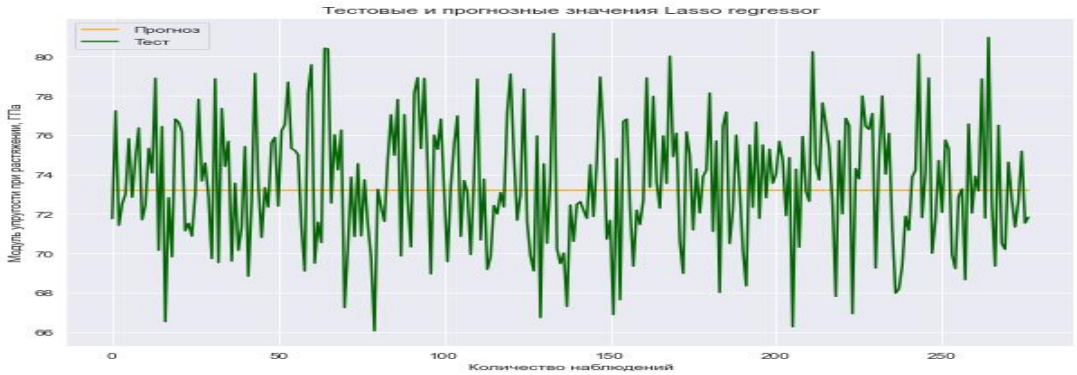


Рисунок 3 - график метода Лассо для модуля упругости, Гпа

### Метрики качества моделей в машинном обучении

В процессе оценки качества моделей машинного обучения применяются различные метрики. Рассмотрим наиболее важные из них.

#### Коэффициент детерминации (R2)

Коэффициент детерминации — это метрика, которая измеряет долю дисперсии целевой переменной, объяснённой моделью.

Интерпретация значений:

- Значения близкие к 1 — модель демонстрирует высокое качество и хорошо объясняет данные

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

- Значения близкие к 0 — качество прогноза не превышает среднего значения целевой переменной
- Отрицательные значения — модель показывает низкую объясняющую способность

### Средняя квадратичная ошибка (MSE)

MSE (Mean Squared Error) — метрика, которая измеряет среднее значение квадратов разностей между предсказанными и фактическими значениями.

Основные характеристики:

- Единицы измерения совпадают с целевой переменной
- Минимальное значение стремится к 0, что указывает на высокое качество модели
- Обеспечивает точную оценку предсказательной способности

### Практическое применение метрик

При оценке качества моделей рекомендуется:

- Использовать комплексный подход с применением нескольких метрик
- Учитывать особенности конкретной задачи
- Проводить анализ результатов в контексте предметной области

### Важные аспекты использования метрик

- Универсальность: не существует идеальной метрики для всех задач
- Специфика: выбор метрики должен соответствовать конкретной задаче
- Интерпретация: необходимо учитывать особенности данных при анализе результатов
- Комплексный подход: рекомендуется комбинировать различные метрики для получения полной картины качества модели

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

```
svr2 = make_pipeline(StandardScaler(), SVR(kernel = 'rbf', C = 500.0, epsilon = 1.0))
#обучаем модель
svr2.fit(x_train_2, np.ravel(y_train_2))
#вычисляем коэффициент детерминации
y_pred_svr2 = svr2.predict(x_test_2)
mae_svr2 = mean_absolute_error(y_pred_svr2, y_test_2)
mse_svr_elast2 = mean_squared_error(y_test_2,y_pred_svr2)
print('Support Vector Regression Results Train:')
print("Test score: {:.2f}".format(svr.score(x_train_2, y_train_2))) # Скор для тренировочной выборки
print('Support Vector Regression Results:')
print('SVR_MAE:', round(mean_absolute_error(y_test_2, y_pred_svr2)))
print('SVR_MAPE: {:.2f}'.format(mean_absolute_percentage_error(y_test_2, y_pred_svr2)))
print('SVR_MSE: {:.2f}'.format(mse_svr_elast2))
print("SVR_RMSE:{:.2f}".format (np.sqrt(mse_svr_elast2)))
print("Test score: {:.2f}".format(svr.score(x_test_2, y_test_2))) # Скор для тестовой выборки
```

```
Support Vector Regression Results Train:
Test score: -503536.43
Support Vector Regression Results:
SVR_MAE: 3
SVR_MAPE: 0.05
SVR_MSE: 19.21
SVR_RMSE:4.38
Test score: -446546.79
```

Рисунок 4 - код для вывода различных метрик для метода опорных векторов

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

## 1.3. Разведочный анализ данных

Перед тем как данные попадут на вход моделей машинного обучения, требуется провести их тщательную обработку и очистку. Дело в том, что необработанные и «грязные» данные часто содержат различные искажения и пропуски, что создаёт серьёзную угрозу для качества моделирования. Использование таких данных может привести к критически неверным результатам анализа. Однако нельзя просто удалять проблемные элементы без должного основания. Поэтому первым шагом необходимо провести детальное изучение исходного набора данных, выявить все проблемные места и только после этого приступать к их обработке. Такой подход позволит сохранить качество данных и обеспечить достоверность последующих моделей.

```
In [23]: a = df.describe()
         a.T
```

Out[23]:

| | count | mean | std | min | 25% | 50% | 75% | max |
|---|---|---|---|---|---|---|---|---|
| **Соотношение матрица-наполнитель** | 1023.0 | 2.930366 | 0.913222 | 0.389403 | 2.317887 | 2.906878 | 3.552660 | 5.591742 |
| **Плотность, кг/м3** | 1023.0 | 1975.734888 | 73.729231 | 1731.764635 | 1924.155467 | 1977.621657 | 2021.374375 | 2207.773481 |
| **модуль упругости, ГПа** | 1023.0 | 739.923233 | 330.231581 | 2.436909 | 500.047452 | 739.664328 | 961.812526 | 1911.536477 |
| **Количество отвердителя, м.%** | 1023.0 | 110.570769 | 28.295911 | 17.740275 | 92.443497 | 110.564840 | 129.730366 | 198.953207 |
| **Содержание эпоксидных групп,%_2** | 1023.0 | 22.244390 | 2.406301 | 14.254985 | 20.608034 | 22.230744 | 23.961934 | 33.000000 |
| **Температура вспышки, С_2** | 1023.0 | 285.882151 | 40.943260 | 100.000000 | 259.066528 | 285.896812 | 313.002106 | 413.273418 |
| **Поверхностная плотность, г/м2** | 1023.0 | 482.731833 | 281.314690 | 0.603740 | 266.816645 | 451.864365 | 693.225017 | 1399.542362 |
| **Модуль упругости при растяжении, ГПа** | 1023.0 | 73.328571 | 3.118983 | 64.054061 | 71.245018 | 73.268805 | 75.356612 | 82.682051 |
| **Прочность при растяжении, МПа** | 1023.0 | 2466.922843 | 485.628006 | 1036.856605 | 2135.850448 | 2459.524526 | 2767.193119 | 3848.436732 |
| **Потребление смолы, г/м2** | 1023.0 | 218.423144 | 59.735931 | 33.803026 | 179.627520 | 219.198882 | 257.481724 | 414.590628 |
| **Угол нашивки** | 1023.0 | 0.491691 | 0.500175 | 0.000000 | 0.000000 | 0.000000 | 1.000000 | 1.000000 |
| **Шаг нашивки** | 1023.0 | 6.899222 | 2.563467 | 0.000000 | 5.080033 | 6.916144 | 8.586293 | 14.440522 |
| **Плотность нашивки** | 1023.0 | 57.153929 | 12.350969 | 0.000000 | 49.799212 | 57.341920 | 64.944961 | 103.988901 |

Рисунок 5 - описательная статистика датасета

Разведочный анализ данных проводится с целью формирования первичных представлений о характеристиках распределений переменных в исходном наборе данных. В процессе анализа оценивается качество исходных данных, выявляются пропуски и выбросы, определяется характер взаимосвязей между переменными. На основе полученных результатов формируются гипотезы о

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

наиболее подходящих моделях машинного обучения для решения поставленной задачи.

Для проведения разведочного анализа применяются различные инструменты и методы: оценивается статистические характеристики датасета, строятся гистограммы распределения для каждой переменной в нескольких вариациях, создаются диаграммы «ящик с усами» в интерактивном формате. Также используются попарные графики рассеяния точек, строится график «квантиль-квантиль», создаются различные варианты тепловых карт. Для каждой переменной проводится описательная статистика, выполняется анализ и полное исключение выбросов в ходе пяти повторных итераций. Осуществляется проверка на наличие пропусков и дубликатов, проводится ранговая корреляция по методам Кендалла и Пирсона.

Такой комплексный подход позволяет получить полное представление о структуре данных и подготовить обоснованные рекомендации для дальнейшего моделирования.

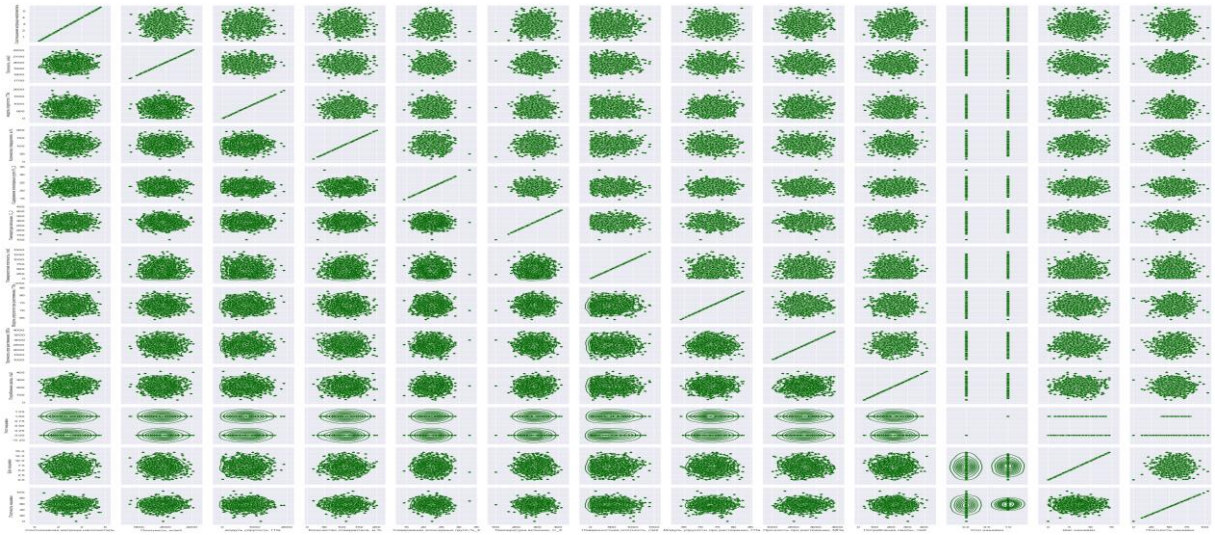

Рисунок 6 - график рассеяния точек

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

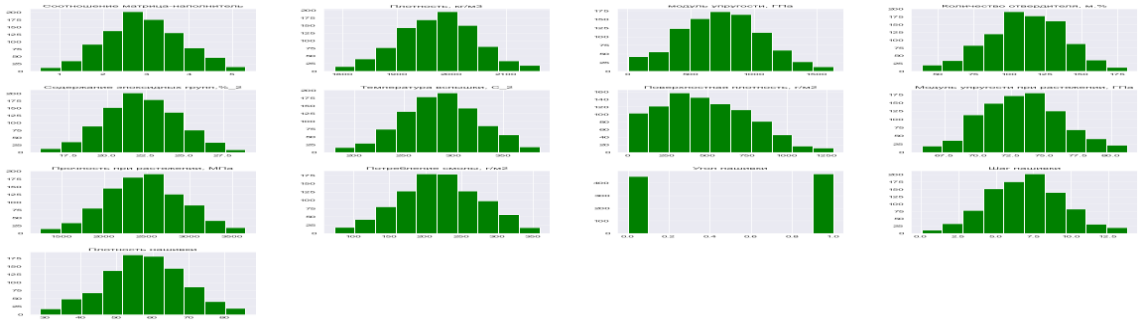

Рисунок 7 - гистограмма распределения

Гистограммы представляют собой важный инструмент визуализации, который используется для исследования распределений частот значений различных переменных в наборе данных. С их помощью можно наглядно оценить характер распределения данных и выявить особенности их распределения.

Проведённый анализ корреляционных связей между переменными показал наличие лишь незначительной зависимости между ними. Это свидетельствует о том, что взаимосвязь между рассматриваемыми параметрами носит скорее случайный характер, а их взаимное влияние минимально.

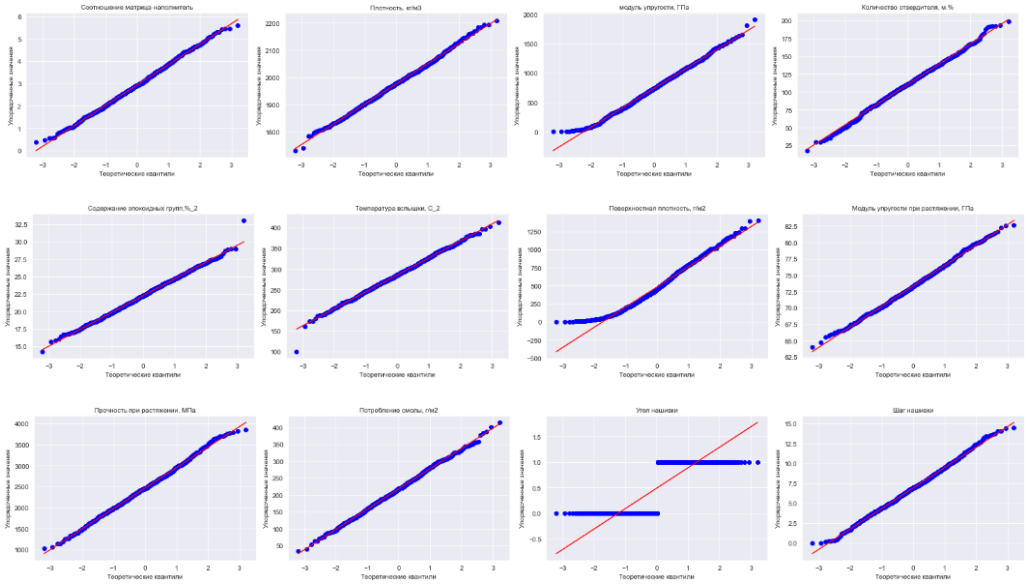

Рисунок 17 - графики «квантиль-квантиль»

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

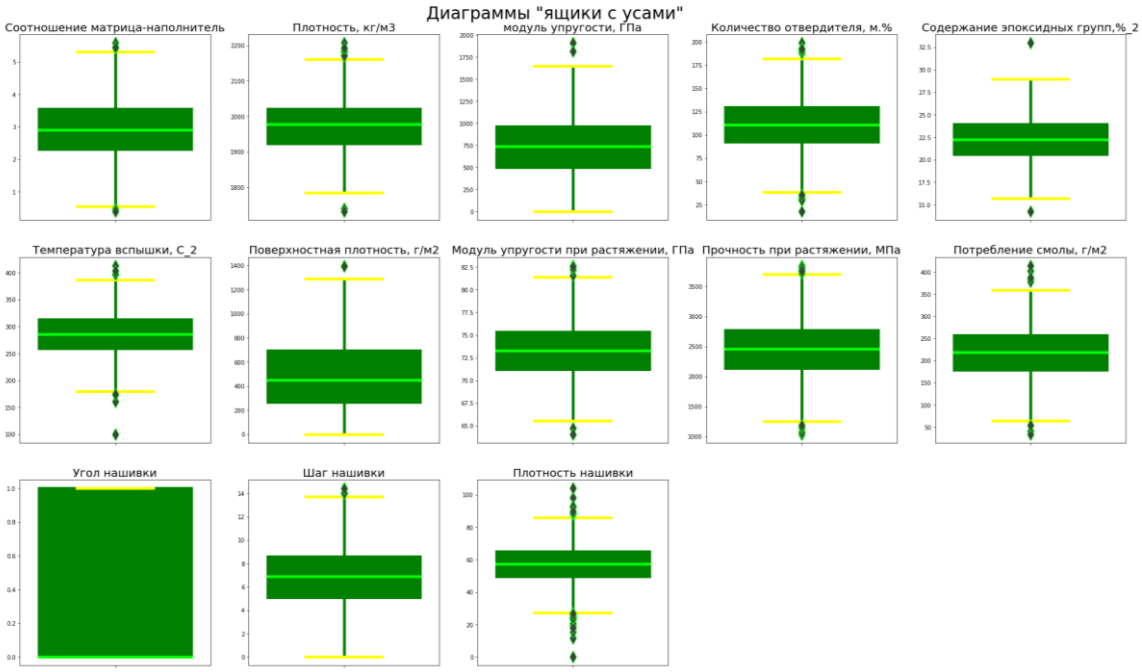


Рисунок 18 - график "ящиков с усами" для всех переменных

После выявления выбросов в данных будет проведён их полный удалением из выборки. Для определения выбросов будут применяться два основных метода: правило трёх сигм и метод межквартильного расстояния. Эти подходы позволят надёжно идентифицировать и исключить значения, существенно отклоняющиеся от общей выборки.

Анализ объединённого датасета показал отсутствие явных зависимостей между переменными. Данный вывод подтверждается результатами визуализации: тепловая карта с матрицей корреляции демонстрирует низкие значения коэффициентов связи между параметрами, а матрицы диаграмм рассеяния не выявляют значимых паттернов взаимосвязей между переменными.

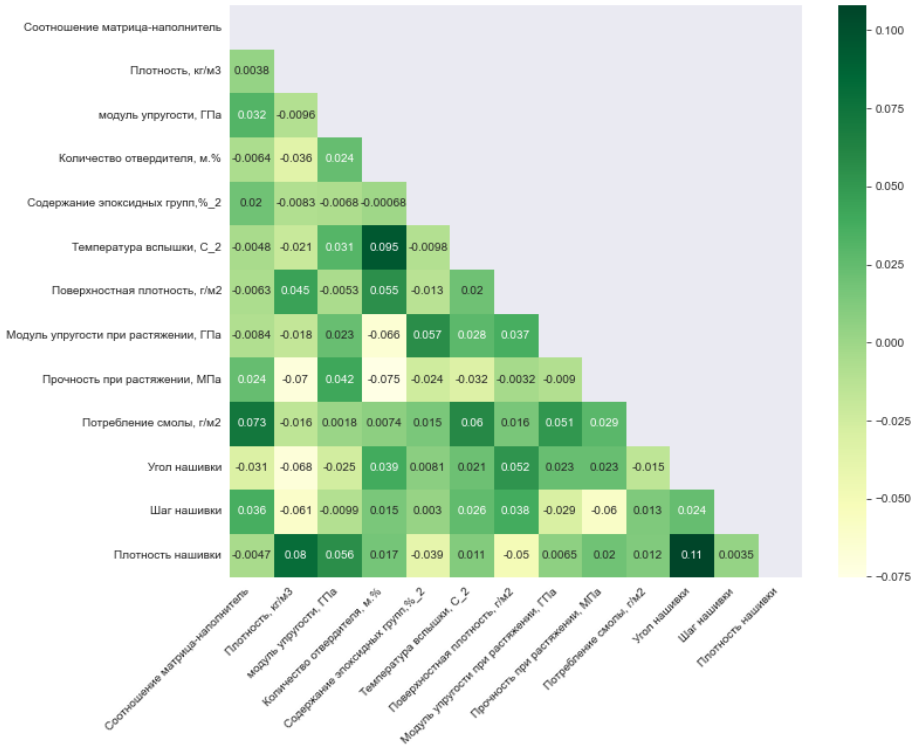


Рисунок 19 - тепловая карта с корреляцией данных

Максимальная корреляция между плотностью нашивки и углом нашивки 0.11, это означает, что зависимости между этими данными нет. Корреляция между всеми параметрами очень близка к 0, корреляционные связи между переменными не наблюдаются.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

# 2. Практическая часть

## 2.1. Предобработка данных

После анализа, приводим столбец "Угол нашивки" к виду «0» и «1». Далее нормализуем значения. Для этого применяем методы MinMaxScaler() и Normalizer(). Второй даёт нам больше выбросов.

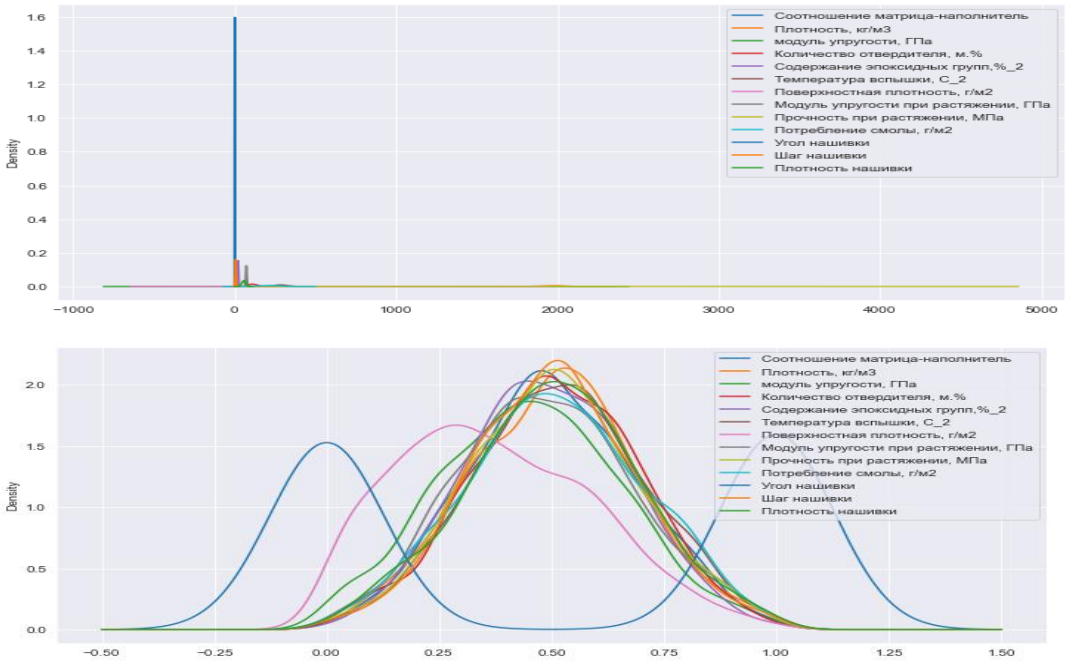


Рисунок 20 - визуализированные данные до и после нормализации

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

## 2.2. Разработка и обучение модели

Разработка и обучение моделей машинного обучения осуществлялась для двух выходных параметров: «Прочность при растяжении» и «Модуль упругости при растяжении» отдельно. Для решения применим все методы, описанные выше.

```
# Проведем поиск  по сетке гиперпараметров с перекрестной проверкой, количество блоков равно 10 (cv = 10), для
# модели случайного леса - Random Forest Regressor - 2

parametrs = { 'n_estimators': [200, 300],
              'max_depth': [9, 15],
              'max_features': ['auto'],
              'criterion': ['mse'] }
grid = GridSearchCV(estimator = rfr, param_grid = parametrs, cv = 10)
grid.fit(x_train_1, y_train_1)
```

```
GridSearchCV(cv=10,
             estimator=RandomForestRegressor(max_depth=7, n_estimators=15,
                                             random_state=33),
             param_grid={'criterion': ['mse'], 'max_depth': [9, 15],
                         'max_features': ['auto'], 'n_estimators': [200, 300]})
```

```
grid.best_params_
```

```
{'criterion': 'mse',
 'max_depth': 15,
 'max_features': 'auto',
 'n_estimators': 200}
```

```
#Выводим гиперпараметры для оптимальной модели
print(grid.best_estimator_)
knr_upr = grid.best_estimator_
print(f'R2-score RFR для прочности при растяжении, МПа: {knr_upr.score(x_test_1, y_test_1).round(3)}')
```

```
RandomForestRegressor(criterion='mse', max_depth=15, n_estimators=200,
                      random_state=33)
R2-score RFR для прочности при растяжении, МПа: 0.963
```

```
#подставим оптимальные гиперпараметры в нашу модель случайного леса
rfr_grid = RandomForestRegressor(n_estimators = 200, criterion = 'mse', max_depth = 15, max_features = 'auto')
#Обучаем модель
rfr_grid.fit(x_train_1, y_train_1)

predictions_rfr_grid = rfr_grid.predict(x_test_1)
#Оцениваем точность на тестовом наборе
mae_rfr_grid = mean_absolute_error(predictions_rfr_grid, y_test_1)
mae_rfr_grid
```

```
67.60356685553326
```

```
new_row_in_mae_df = {'Perpeccop': 'RandomForest_GridSearchCV', 'MAE': mae_rfr_grid}

mae_df = mae_df.append(new_row_in_mae_df, ignore_index=True)
```

Рисунок 21 - поиск гиперпараметров

Разработка модели для каждого параметра и выбранного метода включает последовательное выполнение следующих этапов:

Сначала производится разделение нормализованных данных на две выборки: обучающую (70% данных) и тестовую (30%). Затем осуществляется первичная проверка моделей при использовании стандартных параметров.

На следующем этапе проводится сравнительный анализ с базовой моделью, выдающей среднее значение. Результаты визуализируются с помощью

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

графиков. Качество моделей оценивается по метрике MAE (средняя абсолютная ошибка).

Далее выполняется поиск оптимальных гиперпараметров для оптимизации модели. Основным критерием оценки выступает коэффициент детерминации ($R2$). Процесс оптимизации включает выбор параметров по сетке и проведение кросс-валидации.

После определения оптимальных гиперпараметров они внедряются в модель, которая обучается на тренировочных данных. Затем проводится оценка полученных результатов и их сравнение с исходными стандартными значениями.

Такой поэтапный подход обеспечивает системный процесс разработки и настройки модели, позволяя достичь наилучших показателей качества прогнозирования.

По итогам настройки гиперпараметров модель продемонстрировала некоторое улучшение качества, однако её показатели остались ниже уровня базовой модели. Анализ взаимосвязи между прочностью при растяжении и модулем упругости показал отсутствие линейной зависимости между этими параметрами.

Все применённые модели не смогли достичь требуемого качества прогнозирования, что указывает на неудовлетворительные результаты исследования. Важно отметить, что ключевые характеристики композитных материалов в первую очередь определяются свойствами используемых компонентов. Это обстоятельство необходимо учитывать при разработке новых подходов к моделированию и прогнозированию их поведения.

Полученные результаты указывают на необходимость пересмотра методологии моделирования и поиска альтернативных подходов к решению поставленной задачи.

| | Model | MAE | R2 score |
|---|---|---|---|
| **Модуль упругости при растяжении** | KNeighborsRegressor_upr | 2.517884 | -0.001000 |

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

| | Model | MAE | R2 score |
|---|---|---|---|
| **Прочность при растяжении** | KNeighborsRegressor_pr | 368.126111 | -0.009000 |
| **Модуль упругости при растяжении** | LinearRegression_upr | 2.546419 | -0.021000 |
| **Прочность при растяжении** | LinearRegression_pr | 370.542618 | -0.021000 |
| **Модуль упругости при растяжении** | RandomForestRegressor_upr | 2.597992 | -0.061000 |
| **Прочность при растяжении** | RandomForestRegressor_pr | 370.967356 | -0.013000 |
| **Модуль упругости при растяжении** | MLPRegressor_upr | 2.514920 | -0.000084 |
| **Прочность при растяжении** | MLPRegressor_pr | 367.611428 | -0.009000 |

Таблица 1. Результаты построения и обучения моделей

### 1.1. Тестирование модели

После обучения моделей была проведена оценка точности этих моделей на обучающей и тестовых выборках. В качестве параметра оценки модели использовалась средняя квадратическая ошибка (MSE). Результат неудовлетворительный.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

| | target_var | model_name | MSE | R2 |
|---|---|---|---|---|
| 0 | Модуль упругости при растяжении, ГПа | LinearRegression | 0.02798 | -0.027603 |
| 1 | Прочность при растяжении, МПа | LinearRegression | 0.028816 | -0.011552 |
| 2 | Модуль упругости при растяжении, ГПа | KNeighborsRegressor | 0.02771 | -0.017687 |
| 3 | Прочность при растяжении, МПа | KNeighborsRegressor | 0.02899 | -0.017683 |
| 4 | Модуль упругости при растяжении, ГПа | RandomForestRegressor | 0.027618 | -0.014302 |
| 5 | Прочность при растяжении, МПа | RandomForestRegressor | 0.028608 | -0.004257 |

Рисунок 22 - результат оценки точности по MSE и R2

Out[226]:

| | Перрессор | MAE |
|---|---|---|
| 0 | Support Vector | 78.477914 |
| 1 | RandomForest | 76.589025 |
| 2 | Linear Regression | 61.986894 |
| 3 | GradientBoosting | 64.728717 |
| 4 | KNeighbors | 102.030259 |
| 5 | DecisionTree | 107.158013 |
| 6 | SGD | 181.624450 |
| 7 | MLP | 1808.547264 |
| 8 | Lasso | 69.474334 |
| 9 | RandomForest_GridSearchCV | 67.603567 |
| 10 | KNeighbors_GridSearchCV | 99.281694 |
| 11 | DecisionTree_GridSearchCV | 168.624997 |
| 12 | RandomForest1_GridSearchCV | 2.627032 |
| 13 | KNeighbors1_GridSearchCV | 2.757781 |

Рисунок 23 - результат оценки точности по MAE

При таких результатах можно применить среднее значение переменной в качестве прогнозного.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

## 2.4 **Написать нейронную сеть, которая будет рекомендовать соотношение «матрица – наполнитель».**

Процесс обучения нейронной сети заключается в оптимизации параметров модели таким образом, чтобы минимизировать функционал ошибки. Для реализации этого процесса мы будем использовать класс keras.Sequential, который предоставляет удобный инструмент для создания последовательных нейронных сетей. Этот подход позволяет эффективно выстраивать архитектуру модели путём последовательного добавления слоёв и настройки их параметров.

Перед началом построения сети необходимо подготовить данные и определить ключевые параметры модели. Использование Sequential-класса обеспечивает структурированный подход к разработке нейронной сети, что особенно важно на начальных этапах работы с нейронными сетями.

```
# Сформируем входы и выход для модели

tv = df['Соотношение матрица-наполнитель']
tr_v = df.loc[:, df.columns != 'Соотношение матрица-наполнитель']

# Разбиваем выборки на обучающую и тестовую
x_train, x_test, y_train, y_test = train_test_split(tr_v, tv, test_size = 0.3, random_state = 14)
```

```
# Нормализуем данные

x_train_n = tf.keras.layers.Normalization(axis =-1)
x_train_n.adapt(np.array(x_train))
```

Рисунок 24 - создание нейронной сети

С помощью KerasClassifier находим наилучшие параметры для нашей нейронной сети и построим окончательную нейросеть.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

```
# построение окончательной модели
model = create_model(lyrs=[128, 64, 16, 3], dr=0.05)

print(model.summary())
```

```
Model: "sequential_195"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense_493 (Dense)           (None, 128)               1664

 dense_494 (Dense)           (None, 64)                8256

 dense_495 (Dense)           (None, 16)                1040

 dense_496 (Dense)           (None, 3)                 51

 dropout_195 (Dropout)       (None, 3)                 0

 dense_497 (Dense)           (None, 3)                 12

=================================================================
Total params: 11,023
Trainable params: 11,023
Non-trainable params: 0
_________________________________________________________________
None
```

Рисунок 25 - построение первой нейронной сети

После обучения, оценим модель и посмотрим на потери, задаем функцию для визуализации результатов модели.

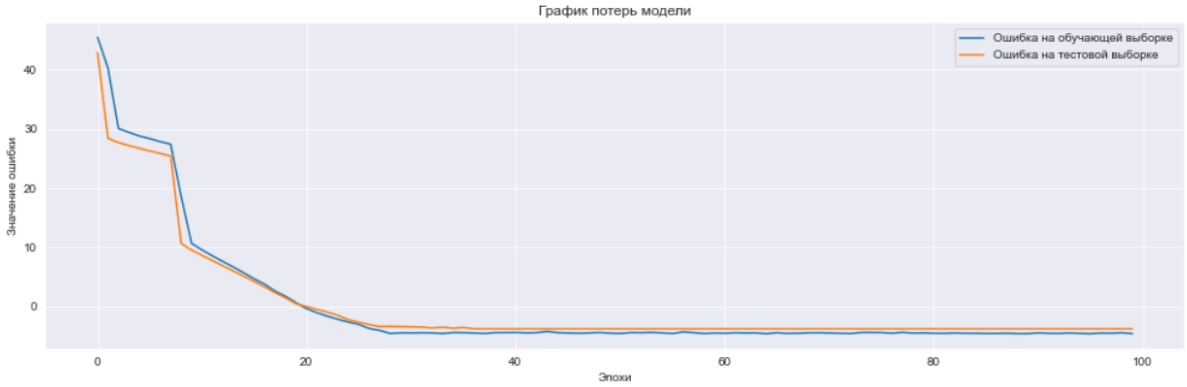


Рисунок 26 - график потерь модели 1

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru


Рисунок 27 - тестовые и прогнозные значения модели 1

Создаем еще одну модель, для получения лучшего результата. После обучения посмотрим на потери, оценим MSE, построим график.

```
# Сконфигурируем модель, зададим слои

model = tf.keras.Sequential([x_train_n, layers.Dense(128, activation='relu'),
                                        layers.Dense(128, activation='relu'), Dropout(0.8),
                                        layers.Dense(128, activation='relu'),
                                        layers.Dense(64, activation='relu'),
                                        layers.Dense(32, activation='relu'),
                                        layers.Dense(16, activation='relu'),
                                        layers.Dense(1)
                             ])

model.compile(optimizer = tf.keras.optimizers.Adam(0.001), loss = 'mean_squared_error', metrics = [tf.keras.metrics.RootMeanSquaredError()])
# Посмотрим на архитектуру модели

model.summary()
```

Рисунок 28 - создание второй модели

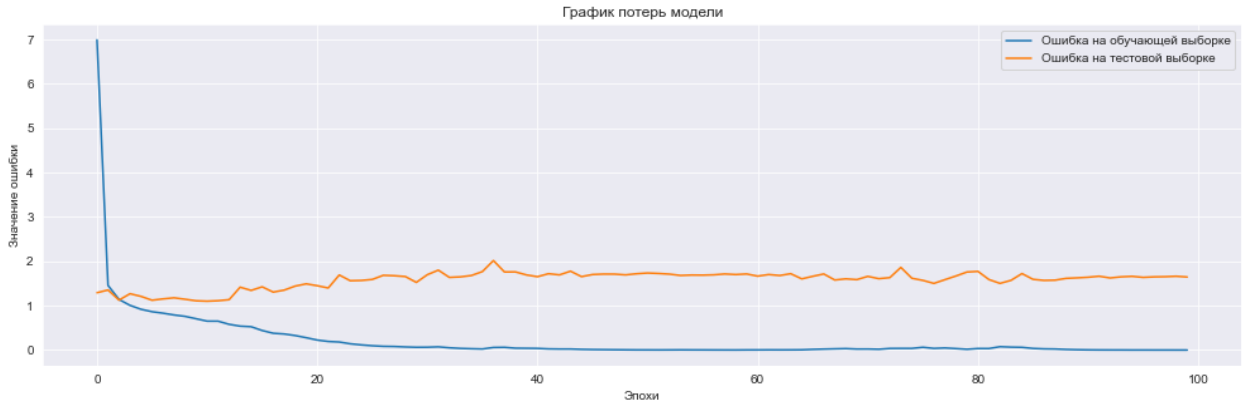


Рисунок 29 - график потерь второй модели

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru


Рисунок 30- тестовые и прогнозные значения второй модели


Рисунок 31 - график прогнозных и настоящих значений

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

## Разработка приложения

Разрабатываем приложение, для результата прогноза для соотношения «матрица-наполнитель». Проверяем его на работоспособность.

Данное приложение — это основной файл Flask, папка templates, с шаблоном html - страницы, папка s_model с сохранённой моделью для данных.

При запуске приложения, пользователь переходит на: http://127.0.0.1:5000/. В открывшемся окне пользователю необходимо ввести в соответствующие ячейки требуемые значения и нажать на кнопку «Готово».

На выходе пользователь получает результат прогноза для значения параметра «Соотношение «матрица – наполнитель»».

```python
@app.route('/', methods=['post', 'get'])

def app_calculation():
    param_lst = []
    message = ''
    if request.method == 'POST':

        # получим данные из наших форм и кладем их в список, который затем передадим функции set_params
         for i in range(1,13,1):
             param = request.form.get(f'param{i}')
             param_lst.append(float(param))

         message = set_params(*param_lst)
```

Рисунок 33 - часть кода приложения

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

## 2.5. **Создание удалённого репозитория и загрузка**

Репозиторий был создан на github.com по адресу:

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

Ноутбук с решением и приложением так же можно найти по адресу:

## 2.6. Заключение

Проведённое исследование позволило сформулировать ключевые выводы относительно анализа данных и прогнозирования свойств композитных материалов. Анализ объединённого датасета показал, что распределение данных приближается к нормальному, однако коэффициенты корреляции между парами признаков практически равны нулю, что свидетельствует об отсутствии значимых линейных взаимосвязей между параметрами.

Методы моделирования, применённые в ходе исследования, не обеспечили получение достоверных прогнозных значений. Модели регрессии продемонстрировали ограниченную эффективность при прогнозировании характеристик композитных материалов. Среди использованных подходов наилучшие результаты были достигнуты методом опорных векторов при прогнозировании модуля упругости при растяжении (ГПа) и лассо-регрессией при оценке прочности при растяжении (МПа).

Исследование выявило невозможность определения соотношения «матрица-наполнитель» на основе имеющихся данных о свойствах материалов. Это обстоятельство не свидетельствует о принципиальной невозможности прогнозирования характеристик композитов с использованием текущего набора данных, но указывает на потенциальные ограничения существующей базы данных, применяемых подходов к прогнозированию и выбранных инструментов анализа.

Для повышения качества прогнозирования требуется комплексный подход, включающий расширение набора входных данных, создание новых результирующих признаков через математические преобразования, релевантные конкретной предметной области, привлечение экспертного мнения специалистов в области материаловедения, проведение

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

дополнительных исследований и формирование междисциплинарной команды исследователей.

Особо следует отметить, что попытки прогнозирования свойств композитных материалов без глубокого понимания основ материаловедения и экспериментального анализа характеристик композитов не приводят к удовлетворительным результатам. Улучшение качества моделей требует внедрения производных показателей и выявления новых уровней взаимосвязей между параметрами.

Учитывая отсутствие значимых корреляций между признаками и результаты применения существующих алгоритмов, можно сделать вывод о том, что решение поставленной задачи либо требует существенно более сложных подходов, либо может оказаться принципиально неосуществимым в рамках текущих методологических ограничений.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

## 2.7. Список используемой литературы и веб ресурсы.

1. Alex Maszański. Метод k-ближайших соседей (k-nearest neighbour): – Режим доступа: https://proglib.io/p/metod-k-blizhayshih-sosedey-k-nearest-neighbour-2021-07-19.

2. Andre Ye. 5 алгоритмов регрессии в машинном обучении, о которых вам сле-дует знать: – Режим доступа: https://habr.com/ru/company/vk/blog/513842/

3. Devpractice Team. Python. Визуализация данных. Matplotlib. Seaborn. Mayavi. - devpractice.ru. 2020. - 412 с.: ил.

4. Абросимов Н.А.: Методика построения разрешающей системы уравнений динамического деформирования композитных элементов конструкций (Учебно-методическое пособие), ННГУ, 2010

5. 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.

6. Грас Д. Data Science. Наука о данных с нуля: Пер. с англ. - 2-е изд., перераб. и доп. - СПб.: БХВ-Петербург, 2021. - 416 с.: ил.

7. Документация по библиотеке keras: – Режим доступа: https://keras.io/api/.

8. Документация по библиотеке matplotlib: – Режим доступа: https://matplotlib.org/stable/users/index.html.

9. Иванов Д.А., Ситников А.И., Шляпин С.Д – Композиционные материалы: учебное пособие для вузов, 2019. 13 с.

10. Краткий обзор алгоритма машинного обучения Метод Опорных Векторов (SVM) – Режим доступа: https://habr.com/ru/post/428503/

11. Ларин А. А., Способы оценки работоспособности изделий из композиционных материалов методом компьютерной томографии, Москва,

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

2013, 148 с.

12. Материалы конференции: V Всероссийская научно-техническая конференция «Полимерные композиционные материалы и производственные технологии нового поколения», 19 ноября 2021 г.

13. Миронов А.А. Машинное обучение часть I ст.9 – Режим доступа: http://is.ifmo.ru/verification/machine-learning-mironov.pdf.

14. Роббинс, Дженнифер. HTML5: карманный справочник, 5-е издание.: Пер. с англ. - М.: ООО «И.Д. Вильямс»: 2015. - 192 с.: ил.

15. Руководство по быстрому старту в flask: – Режим доступа: https://flask-russian-docs.readthedocs.io/ru/latest/quickstart.html.

16. Силен Дэви, Мейсман Арно, Али Мохамед. Основы Data Science и Big Data. Python и наука о данных. – СПб.: Питер, 2017. – 336 с.: ил.

17. Скиена, Стивен С. С42 Наука о данных: учебный курс.: Пер. с англ. - СПб.: ООО "Диалектика", 2020. - 544 с. : ил.

18. Справочник по композиционным материалам: в 2 - х кн. Кн. 2 / Под ред. Дж. Любина; Пер. с англ. Ф. Б. Геллера, М. М. Гельмонта; Под ред. Б. Э. Геллера - М.: Машиностроение, 1988. - 488 с. : ил;

19. Траск Эндрю. Грокаем глубокое обучение. – СПб.: Питер, 2019. – 352 с.: ил.

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

## 1.2. Приложение 1

Подробный план работы:

1. Загружаем и обрабатываем входящие датасеты
   1.1. Удаляем неинформативные столбцы
   1.2. Объединяем датасеты по методу INNER
2. Проводим разведочный анализ данных:
   2.1. Данные в столбце "Угол нашивки» приведём к 0 и 1
   2.2. Изучим описательную статистику каждой переменной - среднее, медиана, стандартное отклонение, минимум, максимум, квартили
   2.3. Проверим датасет на пропуски и дубликаты данных
   2.4. Получим среднее, медианное значение для каждой колонки (по заданию необходимо получить их отдельно, поэтому продублируем их только отдельно)
   2.5. Вычислим коэффициенты ранговой корреляции Кендалла
   2.6. Вычислим коэффициенты корреляции Пирсона
3. Визуализируем наш разведочный анализ сырых данных (до выбросов и нормализации)
   3.1. Построим несколько вариантов гистограмм распределения каждой переменной
   3.2. Построим несколько вариантов диаграмм ящиков с усами каждой переменной
   3.3. Построим гистограмму распределения и диаграмма "ящик с усами" одновременно вместе с данными по каждому столбцу
   3.4. Построим несколько вариантов попарных графиков рассеяния точек (матрицы диаграмм рассеяния)
   3.5. Построим графики квантиль-квантиль
   3.6. Построим корреляционную матрицу с помощью тепловой карты
4. Проведём предобработку данных (в данном пункте только очистка датасета

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

от выбросов)

4.1. Проверим выбросы по 2 методам: 3-х сигм или межквартильных расстояний
4.2. Посчитаем распределение выбросов по каждому столбцу (с целью предотвращения удаления особенностей признака или допущения ошибки)
4.3. Исключим выбросы методом межквартильного расстояния
4.4. Удалим строки с выбросами
4.5. Визуализируем датасет без выбросов, и убедимся, что выбросы еще есть.
4.6. Для полной очистки датасета от выбросов повторим пункты (4.3 – 4.5) ещё 3 раза.
4.7. Сохраняем идеальный, без выбросов датасет
4.8. Изучим чистые данные по всем параметрам
4.9. Визуализируем «чистый» датасет (без выбросов)

5. Проведём нормализацию и стандартизацию (продолжим предобработку данных)
   5.1. Визуализируем плотность ядра
   5.2. Нормализуем данные с помощью MinMaxScaler()
   5.3. Нормализуем данные с помощью Normalizer()
   5.4. Сравним с данными до нормализации
   5.5. Проверим перевод данных из нормализованных в исходные
   5.6. Рассмотрим несколько вариантов корреляции между параметрами после нормализации
   5.7. Стандартизируем данные
   5.8. Визуализируем данные корреляции
   5.9. Посмотрим на описательную статистику после нормализации и после стандартизации
6. Разработаем и обучим нескольких моделей прогноза прочности при

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

растяжении (с 30% тестовой выборки)

6.1. Определим входы и выходы для моделей

6.2. Разобьём данные на обучающую и тестовую выборки

6.3. Проверим правильность разбивки

6.4. Построим модели и найдём лучшие гиперпараметры (задача по заданию):

6.5. Построим и визуализируем результат работы метода опорных векторов

6.6. Построим и визуализируем результат работы метода случайного леса

6.7. Построим и визуализируем результат работы линейной регрессии

6.8. Построим и визуализируем результат работы метода градиентного бустинга

6.9. Построим и визуализируем результат работы метода К ближайших соседей

6.10. Построим и визуализируем результат работы метода деревья решений

6.11. Построим и визуализируем результат работы стохастического градиентного спуска

6.12. Построим и визуализируем результат работы многослойного перцептрона

6.13. Построим и визуализируем результат работы лассо регрессии

6.14. Сравним наши модели по метрике MAE

6.15. Найдём лучшие гиперпараметры для случайного леса

6.16. Подставим значения в нашу модель случайного леса

6.17. Найдём лучшие гиперпараметры для К ближайших соседей

6.18. Подставим значения в нашу модель К ближайших соседей

6.19. Найдём лучшие гиперпараметры метода деревья решений

6.20. Подставим значения в нашу модель метода деревья решений

6.21. Проверим все модели и процессинги и выведем лучшую модель и процессинг

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

7. Разработаем и обучим нескольких моделей прогноза модуля упругости при растяжении (с 30% тестовой выборки)
   7.1. Определим входы и выходы для моделей
   7.2. Разобьём данные на обучающую и тестовую выборки
   7.3. Проверим правильность разбивки
   7.4. Построим модели и найдём лучшие гиперпараметры (задача по заданию):
   7.5. Построим и визуализируем результат работы метода опорных векторов
   7.6. Построим и визуализируем результат работы метода случайного леса
   7.7. Построим и визуализируем результат работы линейной регрессии
   7.8. Построим и визуализируем результат работы метода градиентного бустинга
   7.9. Построим и визуализируем результат работы метода К ближайших соседей
   7.10. Построим и визуализируем результат работы метода деревья решений
   7.11. Построим и визуализируем результат работы стохастического градиентного спуска
   7.12. Построим и визуализируем результат работы многослойного перцептрона
   7.13. Построим и визуализируем результат работы лассо регрессии
   7.14. Сравним наши модели по метрике MAE
   7.15. Найдём лучшие гиперпараметры для случайного леса
   7.16. Подставим значения в нашу модель случайного леса
   7.17. Найдём лучшие гиперпараметры для К ближайших соседей
   7.18. Подставим значения в нашу модель К ближайших соседей
   7.19. Найдём лучшие гиперпараметры метода деревья решений
   7.20. Подставим значения в нашу модель метода деревья решений
   7.21. Проверим все модели и процессинги и выведем лучшую модель и

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

процессинг

8. Нейронная сеть для рекомендации соотношения матрица-наполнитель
   - 8.1. Сформируем входы и выход для модели
   - 8.2. Нормализуем данные
   - 8.3. Построим модель, определим параметры
   - 8.4. Найдем оптимальные параметры для модели
   - 8.5. Посмотрим на результаты
   - 8.6. Повторим шаги 8.4 – 8.5 до построения окончательной модели
   - 8.7. Обучим нейросеть 80/20
   - 8.8. Оценим модель
   - 8.9. Посмотрим на потери модели
   - 8.10. Посмотрим на график результата работы модели
   - 8.11. Посмотрим на график потерь на тренировочной и тестовой выборках
   - 8.12. Сконфигурируем другую модель, зададим слои
   - 8.13. Посмотрим на архитектуру другой модели
   - 8.14. Обучим другую модель
   - 8.15. Посмотрим на потери другой модели
   - 8.16. Посмотрим на график потерь на тренировочной и тестовой выборках
   - 8.17. Зададим функцию для визуализации факт/прогноз для результатов моделей
   - 8.18. Посмотрим на график результата работы модели
   - 8.19. Оценим модель MSE
   - 8.20. Сохраняем вторую модель для разработки веб-приложения для прогнозирования соотношения "матрица-наполнитель" в фреймворке Flask
9. Создаём приложение
   - 9.1. Импортируем необходимые бибилиотеки

Москва, ул. 2-я Бауманская, д.5, стр.1
+7 (495) 187-85-85, do@bmstu.ru
do.bmstu.ru

9.2. Загрузим модель и определим параметры функции

9.3. Получим данные из наших форм и положим их в список

9.4. Укажем шаблон и прототип сайта для вывода

9.5. Запустим приложение

9.6. Откроем http://127.0.0.1:5000/

10. Создание удалённого репозитория и загрузка результатов работы на него.

10.1.

10.2. Создадим README ( )

10.3. Выгрузим все необходимые файлы в репозиторий