МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Московский государственный технический университет имени Н.Э. Баумана

(национальный исследовательский университет)»

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

**по курсу**

«Data Science»

Слушатель Гумяров Рафаэль Алимжанович

Москва, 2022

**СОДЕРЖАНИЕ**

Введение

1. Аналитическая часть

* 1. Постановка задачи
  2. Описание используемых методов
  3. Разведочный анализ данных

2. Практическая часть

* 1. Предобработка данных
  2. Разработка и обучение модели
  3. Тестирование модели
  4. Разработка нейронной сети, которая будет рекомендовать определенные характеристики.
  5. Разработка приложения
  6. Создание удаленного репозитория и загрузка результатов работы на него

Заключение

Список использованной литературы

**ВВЕДЕНИЕ**

**Тема:** Целью данной работы является 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, 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).

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

**Датасет:** <https://drive.google.com/file/d/1B1s5gBlvgU81H9GGolLQVw_SOi-vyNf2/view?usp=sharing>

***1.Аналитическая часть***

**1.1 Постановка задачи**

Целью работы является разработка и обучение ML-моделей для прогнозирования характеристик композитного материала и разработка системы которая будет рекомендовать величину выбранного параметра на основе имеющихся данных. При этом нужно решить три задачи:

* Решить задачу регрессии для прогнозирования показателя:

Модуль упругости при растяжении, Гпа

* Решить задачу регрессии для прогнозирования показателя:

Прочность при растяжении, Мпа

* **Р**азработать рекомендательную систему (задача регрессии) для прогнозирования показателя **:**

Соотношение матрица-наполнитель

Входные данные представляют собой датасет состоящий из двух файлов, которые содержат экспериментально полученные характеристики образцов композитных материалов. Данные файлы были загружены из облачного хранилища (Google Drive) в рабочую среду с последующим их объединением. Объединение файлов было осуществлено по индексу тип объединения INNER.

**1.2 Разведочный анализ данных**

Собрана общая статистика по датасету, использованы методы описательной статистики, в результате чего установлено, что данные содержит:

* + 1022 строки
  + 13 столбцов – показателей
    - 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
  + 1022 строки
  + Данные имеют разную размерность, есть нулевые,
  + Пропуски отсутствуют
  + Выявлена одна дискретная величина
  + первый столбец является индексом (в последствии удален)

Построены гистограммы распределения каждой из переменной и диаграммы «ящик с усами».

Выявлены следующие особенности:

* Гистограммы части параметров имеют смещения и асимметрию (Поверхностная плотность, Температура вспышки, нормального распределение, модуль упругости - необходимо провести центрирование с нормализацией.
* Диаграммы показывают наличие выбросов у всех показателей датасета – необходимо провести их удалить.

1. ***Практическая часть***
   1. **Предобработка данных**

В результате проведенного разведочного анализа данных (Аналитическая часть) принято решение провести центрирование с нормализацией. Для этого буду использовать преобразование PowerTransformer (sklearn.preprocessing) -это семейство параметрических монотонных преобразований, которые применяются для того, чтобы сделать данные более гауссово-подобными.

Для количественной оценки аномалий использую алгоритмом "Изолирующий лес».

Данный алгоритм обнаруживает аномалии, используя изоляцию (насколько далеко точка данных находится от остальных данных), а не моделируя нормальные точки. В результате было выявлено 81 аномалия, которые были удалены. (после очистки, осталось 942 строки).



После чего проведена повторная визуализация данных (отрисованы диаграммы «ящик с усами») и выявлено большое остаточное количество выбросов.

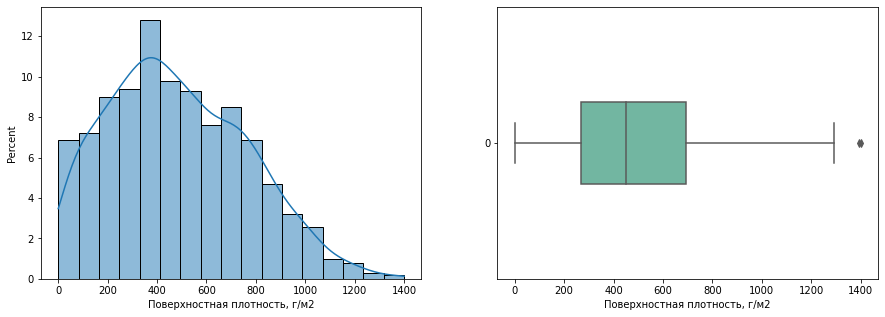
Для их исключения использован метод 3-х сигм.

Правило трёх сигм: отклонение значения нормально распределённой случайной величины X от её математического ожидания М(х) не превосходит утроенного среднеквадратического отклонения σ с вероятностью около 0,9973. В результате работы алгоритма удалено 72 выброса. (после очистки, осталось 870 строки).

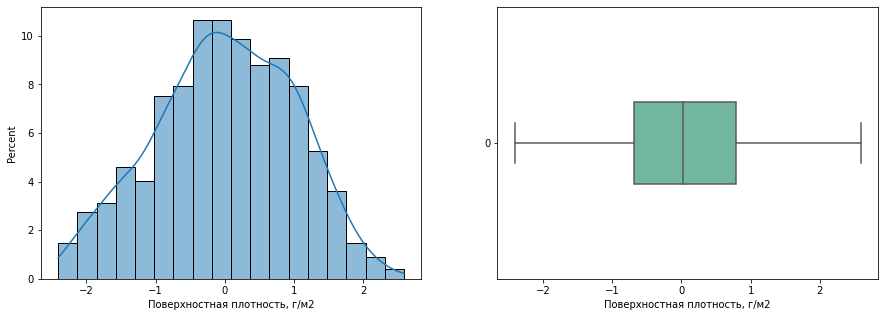
После чего проведена повторная визуализация данных (отрисованы диаграммы «ящик с усами»).

Результат проделанной работы представлен на примере параметра «Поверхностная плотность» на картинках ниже:

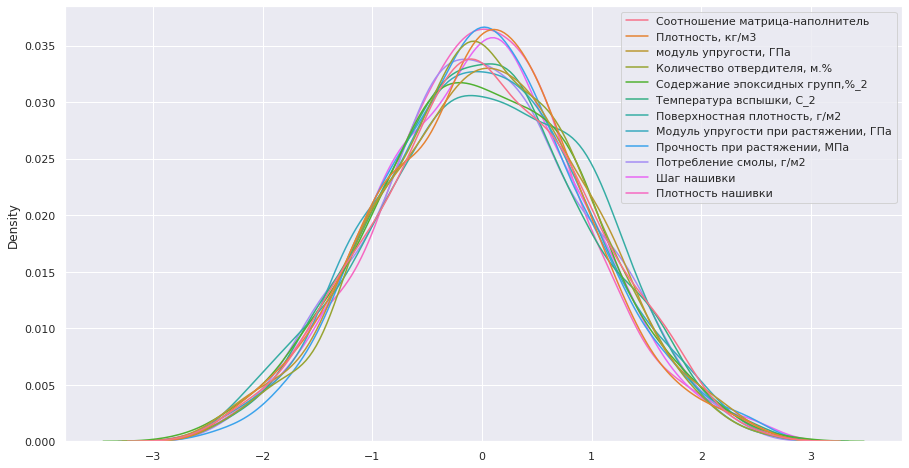
До:



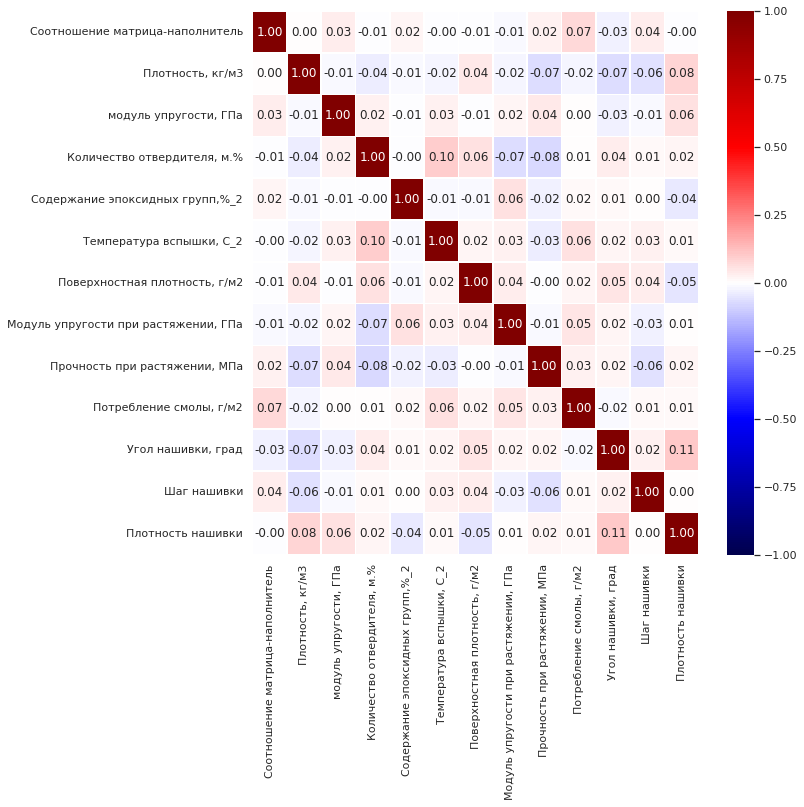
После:



Для того что бы удостоверится в сопоставимости параметров были построены графики распределения для каждого параметра: параметры сопоставимы как по диапазону так и по амплетуде.



Для поиска взаимосвязи параметров была построена тепловая карта коэффициентов корреляции, которая не выявила резко выраженных взаимосвязей параметров.



Так же построены попарные графики рассеяния точек, которые показали отсутствие выраженных корреляций параметров.

**2.2 Разработка и обучение модели**

На данном этапе необходимо разработать несколько ML-моделей для прогноза показателей:

* "Модуль упругости при растяжении",
* "Прочность при растяжении".

Задача прогноза показателей сводится к решению задачи регрессии в машинном обучении - предсказанию одного параметра (Y) по известному параметру X, где X — набор параметров, характеризующий наблюдение.

Необходимо выбрать и обучить несколько моделей для прогноза вышеуказанных характеристик.

Выбраны следующие модели:

* Линейная регрессия (LinearRegression) – применялся для прогноза параметра «Модуль упругости при растяжении».
* Деревья решений (DT) (DecisionTreeRegressor) - применялся для прогноза параметра «Модуль упругости при растяжении».
* K-ближайших соседей (KNN) (KNeighborsRegressor) - применялся для прогноза «Прочность при растяжении».

*Линейная регрессия (LinearRegression)*

**Линейная регрессия** – одна из важнейших и широко используемых техник регрессии. Эта самый простой метод регрессии. Одним из его достоинств является лёгкость интерпретации результатов. Линейная регрессия подгоняет линейную модель к минимизации остаточной суммы квадрата, наблюдаемого целевого признака в наборе данных и предсказанного целевого признака по линейной аппроксимации (метод наименьших квадратов).

Оценки коэффициентов для обыкновенных наименьших квадратов полагаются на независимость функций. В противном случае метод наименьших квадратов становится очень чувствительной к случайным ошибкам, что приводит к большой дисперсии.

**K-ближайших соседей (KNN)**

KNN - это один из самых простых алгоритмов классификации, также иногда используемый в задачах регрессии. В случае использования метода для [регрессии](https://ru.wikipedia.org/wiki/%D0%A0%D0%B5%D0%B3%D1%80%D0%B5%D1%81%D1%81%D0%B8%D1%8F_(%D0%BC%D0%B0%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0)), объекту присваивается среднее значение по K ближайшим к нему объектам, значения которых уже известны. Метод взят для сравнения с классическими алгоритмами применяемыми для задач регрессии.

В модели KNN гиперпараметрами обычно выступают:

* Число соседей (n\_neighbors);
* Весовая функция (uniform -однородные веса, 'distance взвешивание точек обратно их расстоянию)

**Деревья решений (DT)**

Деревья решений (DT) — это непараметрический контролируемый метод обучения, используемый для классификации и регрессии . Цель состоит в том, чтобы создать модель, которая предсказывает значение целевой переменной, изучая простые правила принятия решений, выведенные из характеристик данных. Дерево можно рассматривать как кусочно-постоянное приближение. Одним из плюсов данного метода является не большая требовательность к подготовке данных. Деревья решений могут применяться к задачам регрессии с помощью класса DecisionTreeRegressor

**2.3 Тестирование модели**

Для решения поставленных задач буду использовать библиотеку Scikit-Learn У Scikit-learn довольно удобный API. Он предоставляет множество моделей, и все они имеют функции fit и predict. Можно вызвать fit с нашими данными для обучения модели, а затем использовать predict для получения предсказанных значений на основе новых данных. Scikit-learn также предоставляет множество метрик, которые вы можете использовать для оценки.

Работу буду проводить на разделенных данных - на тестовую и тренировочную выборки в отношении 30/70.

Целевой показатель - колонка Прочность при растяжении, МПа. = «Y», Входные данные - подготовленный на предыдущих этапах набор данных (за вычетом целевого показателя) = «X»

*Решение задачи с помощью метода линейной регрессии дало следующие результаты*:

* Коэффициент детерминации: 0.013250267914889036
* Средняя квадратичная ошибка (MSE): 0.9083022208290894
* Средняя абсолютная ошибка (MAE): 0.782541147645433

**Данный результаты являются неудовлетворительным**.

*Решение задачи с помощью метода KNN дало следующие результаты*:

* Коэффициент детерминации: -0.09964254060617073
* Средняя квадратичная ошибка (MSE): 1.8322662889272823
* Средняя абсолютная ошибка (MAE): 1.09029571296525

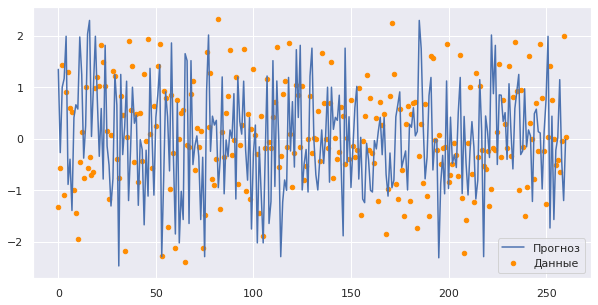
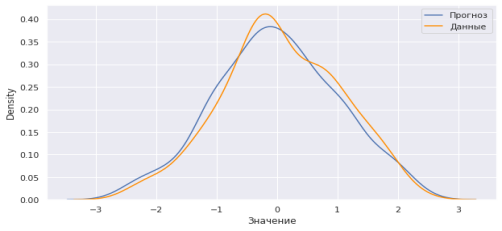
**Данный результаты являются неудовлетворительным**.

* *Решение задачи с помощью метода Деревья решений (DT)*

*дало следующие результаты*:

* Коэффициент детерминации: 1.0
* Средняя квадратичная ошибка (MSE): 1.8322662889272823
* Средняя абсолютная ошибка (MAE): 1.09029571296525

**Данный результаты являются приемлемым**.

**2.4 Разработка нейронной сети, которая будет рекомендовать определенные характеристики**

Поставленная задача регресси сводится к разработке нейронной сети -многослойного персептрона.

Многослойный персептрон (MLP) — это алгоритм обучения с учителем, который изучает функцию f(⋅):Rm→Ro обучением на наборе данных (где m — количество измерений для ввода и o- количество размеров для вывода).

MLP состоит как минимум из трех [слоев](https://translated.turbopages.org/proxy_u/en-ru.ru.a69d8179-62ad783d-977358f1-74722d776562/https/en.wikipedia.org/wiki/Layer_(deep_learning)) узлов: входного [слоя](https://translated.turbopages.org/proxy_u/en-ru.ru.a69d8179-62ad783d-977358f1-74722d776562/https/en.wikipedia.org/wiki/Layer_(deep_learning)), скрытого [слоя](https://translated.turbopages.org/proxy_u/en-ru.ru.a69d8179-62ad783d-977358f1-74722d776562/https/en.wikipedia.org/wiki/Layer_(deep_learning)) и выходного [слоя](https://translated.turbopages.org/proxy_u/en-ru.ru.a69d8179-62ad783d-977358f1-74722d776562/https/en.wikipedia.org/wiki/Layer_(deep_learning)). За исключением входных узлов, каждый узел представляет собой нейрон, который использует нелинейную [функцию активации](https://translated.turbopages.org/proxy_u/en-ru.ru.a69d8179-62ad783d-977358f1-74722d776562/https/en.wikipedia.org/wiki/Activation_function). MLP использует метод [контролируемого обучения](https://translated.turbopages.org/proxy_u/en-ru.ru.a69d8179-62ad783d-977358f1-74722d776562/https/en.wikipedia.org/wiki/Supervised_learning), называемый [обратным](https://translated.turbopages.org/proxy_u/en-ru.ru.a69d8179-62ad783d-977358f1-74722d776562/https/en.wikipedia.org/wiki/Backpropagation) распространением для обучения. Его многослойность и нелинейная активация отличают MLP от линейного [персептрона](https://translated.turbopages.org/proxy_u/en-ru.ru.a69d8179-62ad783d-977358f1-74722d776562/https/en.wikipedia.org/wiki/Perceptron). Он может различать данные, которые не являются [линейно разделимыми](https://translated.turbopages.org/proxy_u/en-ru.ru.a69d8179-62ad783d-977358f1-74722d776562/https/en.wikipedia.org/wiki/Linear_separability).

Многослойный персептрон:

keras.layers.Dense(22, activation=tf.nn.relu,   input\_shape=(x\_train.shape[1],)),

keras.layers.Dense(22, activation=tf.nn.relu),

keras.layers.Dense(11, activation=tf.nn.relu),

keras.layers.Dense(1, activation=tf.nn.sigmoid)  ])

Гиперпараметры модели:

* - количество скрытых слоев
* - количество нейронов на слое
* - активационная функция
* - количество нейронов на выходном слое
* - оптимизатор
* - метрика оценки качества
* - количество эпох

*Решение задачи с помощью* **нейронной сети** *дало следующие результаты*:

Оценка модели: [0.9562991261482239, 0.7879415154457092]

