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 Мациевская С.Г. (ФИО)

Москва, 2022

1. *Аналитическая часть*
   1. *Постановка задачи.*

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).

* 1. *Описание используемых методов*

В качестве входящих данных было предложены два excel файла.

X-bp.xlsx, 1023 строки, 11 колонок.

X\_nup.xlsx, 1040 строки, 4 колонки.

В обоих датасетах первая колонка – это нумерация, она не несет значения, поэтому удаляем их.

Было дано задание объединить файлы по типу INNER.

Представим информацию о датасете после объединения:

<class 'pandas.core.frame.DataFrame'>

Int64Index: 1023 entries, 0 to 1022

Data columns (total 13 columns):

# Column Non-Null Count Dtype

--- ------ -------------- -----

0 Соотношение матрица-наполнитель 1023 non-null float64

1 Плотность, кг/м3 1023 non-null float64

2 модуль упругости, ГПа 1023 non-null float64

3 Количество отвердителя, м.% 1023 non-null float64

4 Содержание эпоксидных групп,%\_2 1023 non-null float64

5 Температура вспышки, С\_2 1023 non-null float64

6 Поверхностная плотность, г/м2 1023 non-null float64

7 Модуль упругости при растяжении, ГПа 1023 non-null float64

8 Прочность при растяжении, МПа 1023 non-null float64

9 Потребление смолы, г/м2 1023 non-null float64

10 Угол нашивки, град 1023 non-null float64

11 Шаг нашивки 1023 non-null float64

12 Плотность нашивки 1023 non-null float64

dtypes: float64(13)

memory usage: 111.9 KB

Все столбцы содержать данные типа float64, нет необходимости производить преобразование.

В данных нет пропусков, то есть нет необходимость думать, как заполнить пропущенные или некорректные данные.

Дубликатов в данных тоже нет.

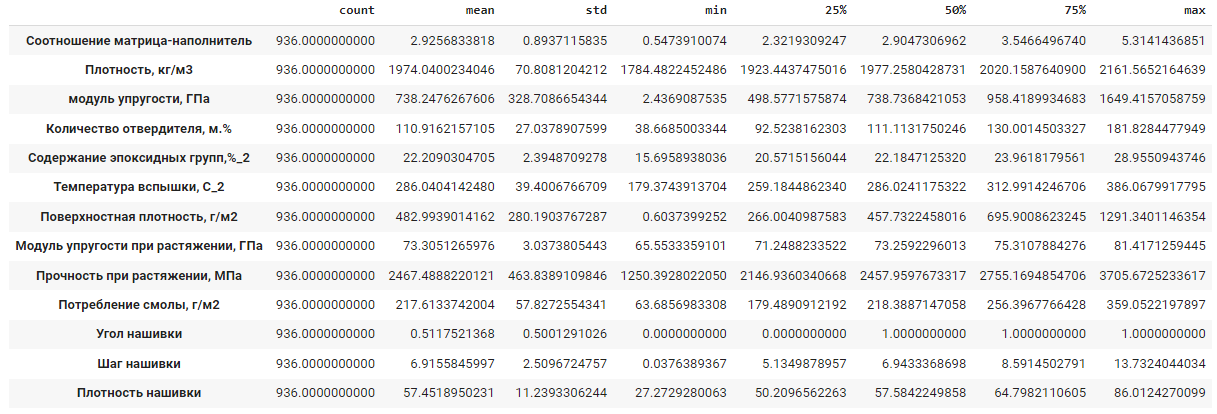
Проверив количество уникальных значений в каждом столбце, видим, что в столбце «Угол нашивки» только 2 значения – 0 и 90 градусов.

Приведем данные в этом столбце к int и преобразуем эти данные в 0 и 1 соответственно для облегчения работы при построении моделей.

В задании было предложено получить среднее, медианное значение для каждой колонки.



* 1. *Разведочный анализ данных*

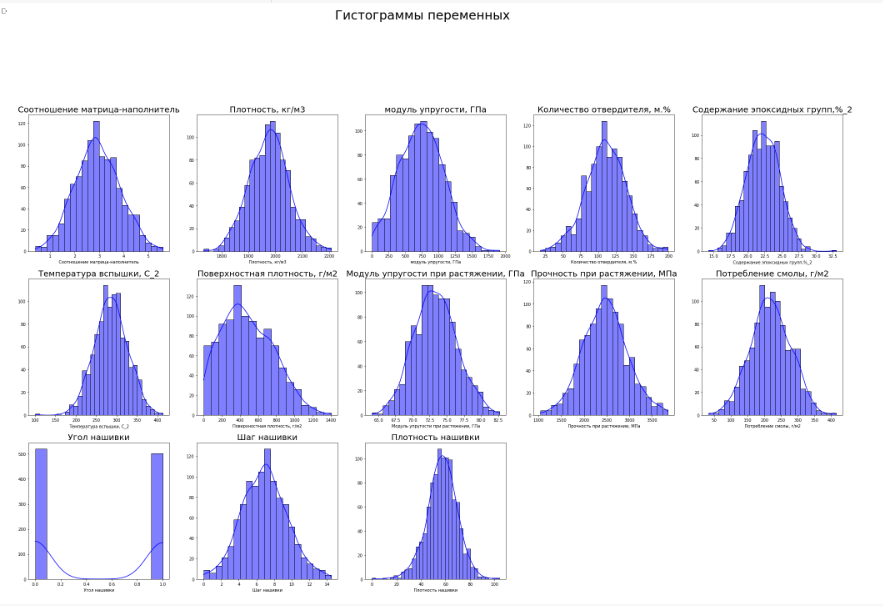
Прежде чем применять те или иные методы обучения, нам необходимо удостовериться, что они применимы к текущему датасету. Раздел описательной статистики включает в себя проверку на нормальность распределения и определение прочих статистических метрик.

Основываясь на данных описательной статистики можно сделать вывод, что разброс в значения довольно значительный и будет разумно данные стандартизировать.

Визуализируем данные, сделав гистограммы распределения по всем колонкам.

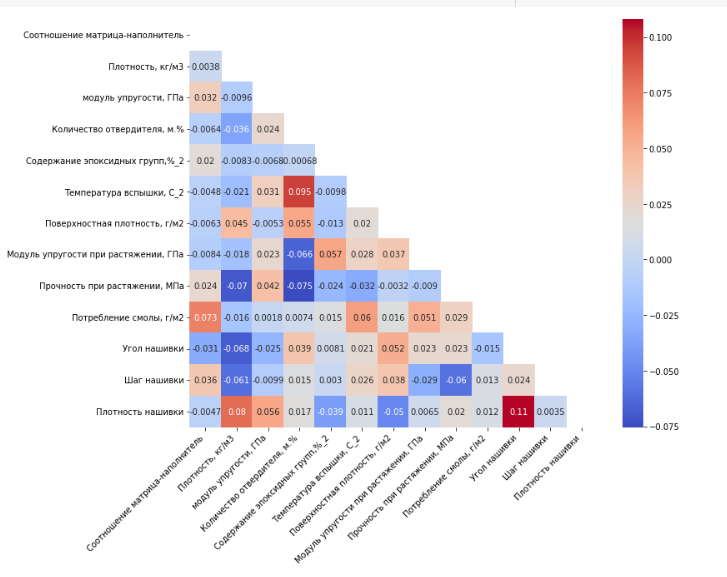
Данные стремятся к нормальному распределению почти во всех столбцах. Исключение – это столбец с углом нашивки, где всего два значения.

И также распределение ненормально у признака Поверхностная плотность, что подтвердилось тестом Шапиро-Уилка.



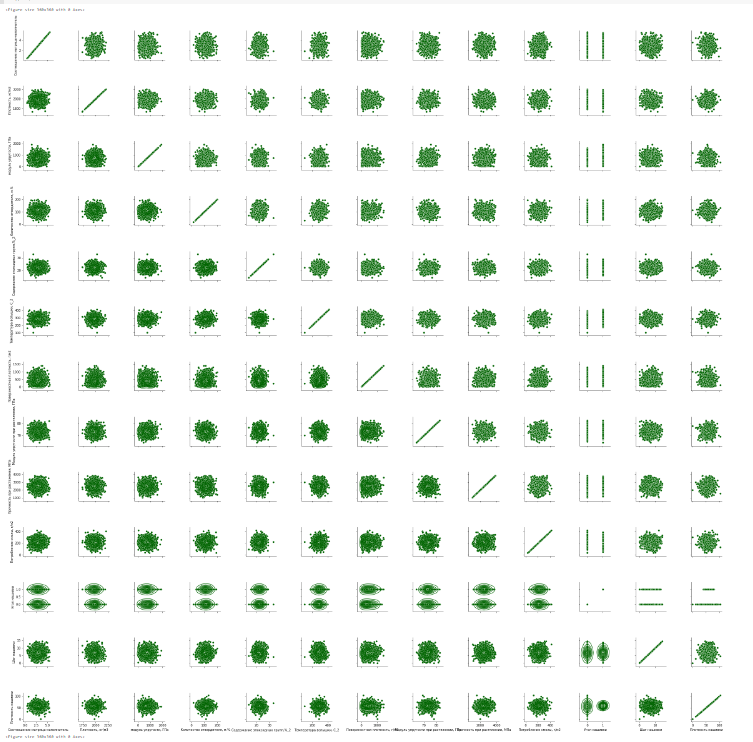
Проверим есть ли зависимости между признаками и между признаками и целевыми переменными.

Визуализируем корреляционную матрицу с помощью тепловой карты.



Построим также парные scatterplots.

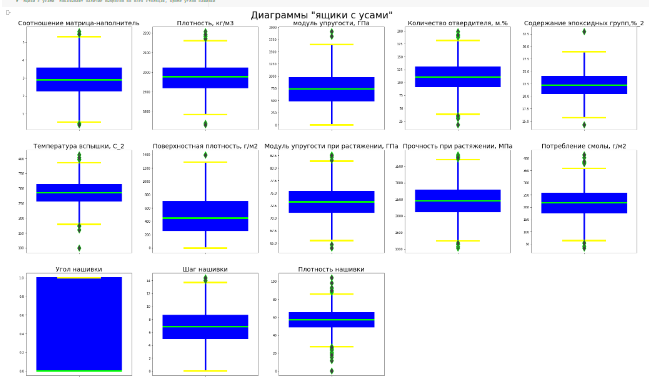
С их помощью можно проверить, есть ли видимая зависимость, возможно не линейная, которую heatmap могла не отобразить.



Можно сделать вывод, что корреляция между признаками незначительна, так что мы не можем исключить никакие из них.

Корреляция между целевыми переменными и признаками также незначительная, так что сложно будет подобрать модель.

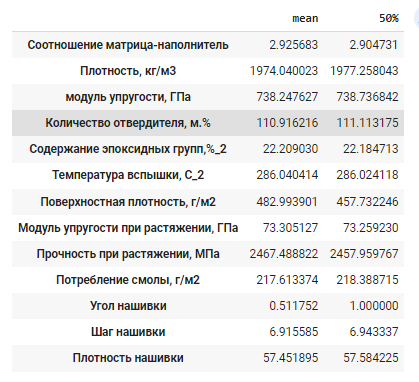
Для визуализации выбросов построим диаграммы boxplot.



Наблюдаем выбросы во всех стоблцах, кроме Угол нашивки.

Для удаления выбросов воспользуемся методом межквартильных расстояний. Он может использоваться для определения выбросов путем определения значений выборки ниже 25-го процентиля или выше 75-го процентиля.

Посмотрим на средние и медианные значения датасета после выброса, убедимся, что они остались в тех пределах, чтобы было до удаления выбросов.

 До удаления выбросов После удаления выбросов

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

Различные методы масштабирования объектов влияют на предсказательную способность моделей, в основном, линейной регрессии, конечно.

В машинном обучении [масштабирование объектов](https://translated.turbopages.org/proxy_u/en-ru.ru.d4268d40-63a267ad-ef9f84ad-74722d776562/https/www.baeldung.com/cs/feature-scaling) относится к помещению значений объектов в один и тот же диапазон. Масштабирование чрезвычайно важно для алгоритмов, учитывающих расстояния между наблюдениями, такими как k-ближайшие соседи. С другой стороны, на алгоритмы, основанные на правилах, такие как деревья решений, масштабирование объектов не влияет.

Метод масштабирования данных заключается в том, чтобы втиснуть их в заранее определенный интервал. При[нормализации](https://translated.turbopages.org/proxy_u/en-ru.ru.d4268d40-63a267ad-ef9f84ad-74722d776562/https/www.baeldung.com/cs/batch-normalization-cnn#normalization) мы сопоставляем минимальное значение признака с 0, а максимальное - с 1. Следовательно, значения признаков отображаются в диапазоне [0, 1].

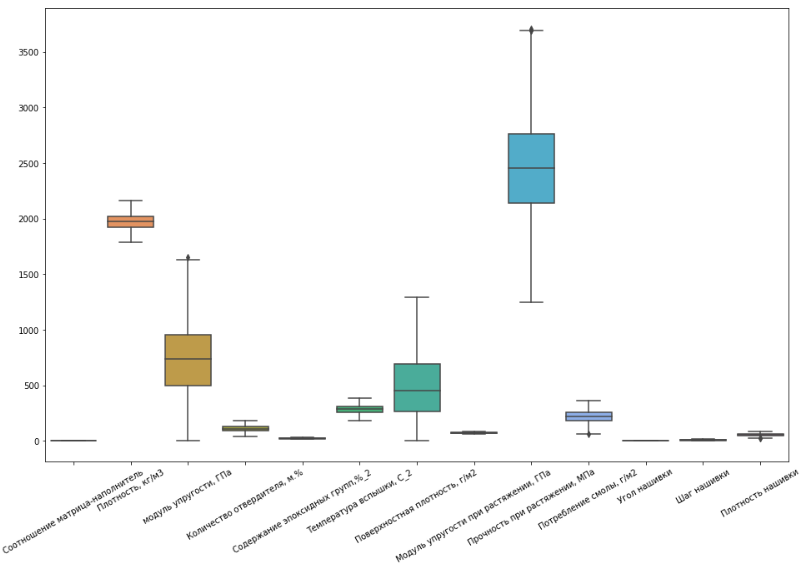
При [стандартизации](https://translated.turbopages.org/proxy_u/en-ru.ru.d4268d40-63a267ad-ef9f84ad-74722d776562/https/www.baeldung.com/cs/normalizing-inputs-artificial-neural-network#2-standardization) мы не вводим данные в определенный диапазон. Вместо этого мы преобразуем, чтобы получить среднее значение 0 и стандартное отклонение 1:

Это не только помогает с масштабированием, но и централизует данные.

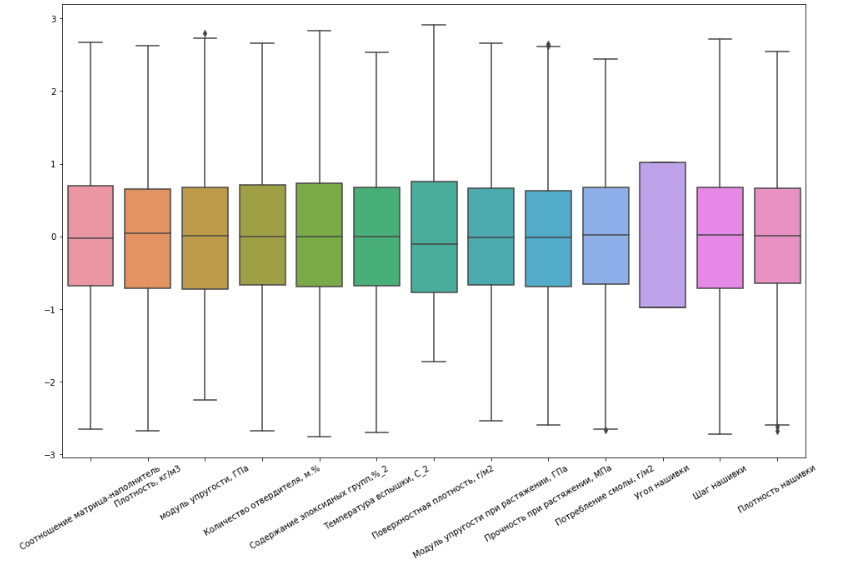
В целом, стандартизация в большинстве случаев более подходит, чем нормализация.

Преобразуем датасет, и представим графики до и после стандартизации.

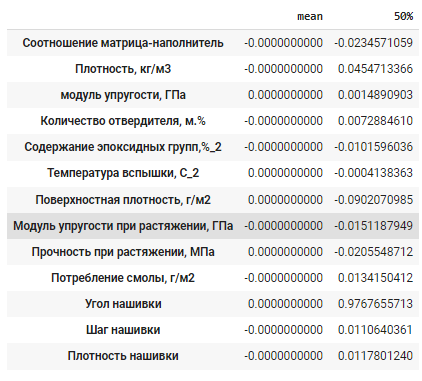
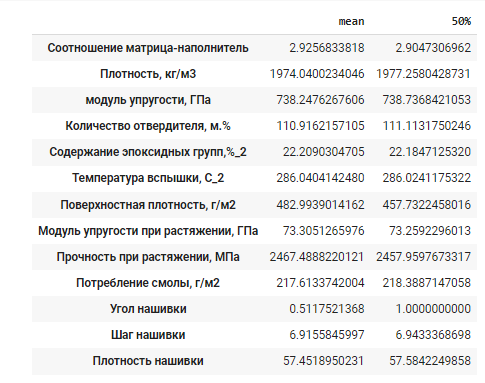
До



После



Укажем максимальные и минимальные значения



* 1. *Разработка и обучение модели*

Так как наши данные размечены, то к ним применимы задачи обучения с учителем: классификация и регрессия.

Классификация задач прогнозного моделирования отличается от задач регрессионного прогнозного моделирования.

* Классификация - это задача прогнозирования метки дискретного класса.
* Регрессия - это задача прогнозирования непрерывного количества.

В нашем случае необходимо сделать предсказание непрерывной величины, значит нам нужна задача регрессии.

Для прогноза модуля упругости при растяжении и прочности при растяжении будут использоваться следующие модели: линейная регрессия, дерево решений, метод случайного леса, метод ближайших соседей, метод градиентного бустинга.

Линейная регрессия - это алгоритм машинного обучения, основанный на обучении под наблюдением. Он выполняет задачу регрессии. Регрессионные модели представляют собой целевое значение прогноза, основанное на независимых переменных. В основном используется для выяснения взаимосвязи между переменными и прогнозирования.

## **Преимущества и недостатки методики линейно регрессии**

Преимущества

1. Линейная регрессия проста в реализации и легче интерпретировать выходные коэффициенты.
2. Когда вы знаете, что связь между независимой и зависимой переменными имеет линейную зависимость, этот алгоритм лучше всего использовать из-за его меньшей сложности по сравнению с другими алгоритмами.
3. Линейная регрессия подвержена чрезмерной подгонке, но ее можно избежать, используя некоторые методы уменьшения размерности, методы регуляризации (L1 и L2) и перекрестной проверки

Недостатки

1. С другой стороны, в технике линейной регрессии выбросы могут оказывать огромное влияние на регрессию, а границы в этой технике линейны.
2. С другой стороны, линейная регрессия предполагает линейную зависимость между зависимыми и независимыми переменными. Это означает, что предполагается, что между ними существует прямая зависимость. Он предполагает независимость между атрибутами.
3. Но тогда линейная регрессия также рассматривает взаимосвязь между средним значением зависимых переменных и независимыми переменными. Так же, как среднее значение не является полным описанием одной переменной, линейная регрессия не является полным описанием взаимосвязей между переменными.

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |

Линейная регрессия - отличный инструмент для анализа взаимосвязей между переменными, но она не рекомендуется для большинства практических приложений, поскольку она чрезмерно упрощает реальные проблемы, предполагая линейную взаимосвязь между переменными.

Дерево решений можно представить как карту возможных результатов из ряда взаимосвязанных выборов. Это помогает сопоставить возможные действия, основываясь на их стоимости (затратах), вероятности и выгоде.

Как понятно из названия, для этого используют модель принятия решений в виде дерева. Такие древовидные схемы могут быть полезны и в процессе обсуждения чего-либо, и для составления алгоритма, который математически определяет наилучший выбор.

## **Преимущества и недостатки методики дерева решений**

**Преимущества метода**:

1. Деревья решений создаются по понятным правилам, они просты в применении и интерпретации.
2. Можно обрабатывать как непрерывные, так и качественные (дискретные) переменные.
3. Можно работать с пропусками в данных, деревья решений позволяют заполнить пустое поле наиболее вероятным значением.
4. Помогают определить, какие поля больше важны для прогнозирования или классификации.

**Недостатки метода**:

1. Есть вероятность ошибок в задачах классификации с большим количеством классов и относительно небольшим числом примеров для обучения.
2. Нестабильность процесса: изменение в одном узле может привести к построению совсем другого дерева, что связано с иерархичностью его структуры.
3. Процесс «выращивания» дерева решений может быть довольно затратным с точки зрения вычислений, ведь в каждом узле каждый атрибут должен раскладываться до тех пор, пока не будет найден наилучший вариант решения или разветвления. В некоторых алгоритмах используются комбинации полей, в таком случае приходится искать оптимальную комбинацию по «весу» коэффициентов. Алгоритм отсечения (или «обрезки») также дорогостоящий, так как необходимо сформировать и сравнить большое количество потенциальных ветвей.

## **Преимущества и недостатки методики случайного леса**

**Преимущества метода**:

1. Он решает проблему переобучения путем усреднения или объединения результатов разных деревьев решений.
2. Случайные леса лучше работают для большого диапазона элементов данных, чем одно дерево решений.
3. Случайный лес имеет меньшую дисперсию, чем одно дерево решений.
4. Случайные леса очень гибкие и обладают очень высокой точностью.
5. Масштабирование данных не требует алгоритма случайного леса. Он поддерживает хорошую точность даже после предоставления данных без масштабирования.
6. Масштабирование данных не требует алгоритма случайного леса. Он поддерживает хорошую точность даже после предоставления данных без масштабирования.

**Недостатки метода**:

1. Сложность является основным недостатком алгоритмов случайного леса.
2. Построение случайных лесов намного сложнее и отнимает много времени, чем деревья решений.
3. Для реализации алгоритма случайного леса требуется больше вычислительных ресурсов.
4. Это менее интуитивно понятно в случае, когда у нас есть большая коллекция деревьев решений .
5. Процесс прогнозирования с использованием случайных лесов занимает очень много времени по сравнению с другими алгоритмами.

## **Преимущества и недостатки методики линейно регрессии**

**Преимущества метода**:

1. Алгоритм прост и легко реализуем.
2. Не чувствителен к выбросам.
3. Нет необходимости строить модель, настраивать несколько параметров или делать дополнительные допущения.
4. Алгоритм универсален. Его можно использовать для обоих типов задач: классификации и регрессии.

**Недостатки метода**:

1. Алгоритм работает значительно медленнее при увеличении объема выборки, предикторов или независимых переменных.
2. Из аргумента выше следуют большие вычислительные затраты во время выполнения.
3. Всегда нужно определять оптимальное значение **k**.

## **Градиентная регрессия**

* Алгоритм Gradient Boosting представляет собой создание леса из фиксированного числа деревьев решений, которые называются слабыми обучаемыми или слабыми прогностическими моделями. Эти деревья решений имеют фиксированный размер или глубину.
* Повышение градиента начинается со среднего значения целевых значений и добавляет прогноз/результат/вклад из последующих деревьев, уменьшая их с помощью так называемой скорости обучения.
* Деревья решений или оценщики обучены прогнозировать отрицательный градиент выборок данных.
* Алгоритм Gradient Boosting Regression используется для подбора модели, которая предсказывает непрерывное значение.
  1. *Тестирование модели*

Представим показатели моделей на тренировочной и тестовой выборках.

#### **Прогноз прочности при растяжении**

Linear Regression Results Train:

Train score: 0.03

Linear Regression Results:

Test score: -0.05

Decision Tree Regressor Results Train:

Test score: 1.00

Decision Tree Regressor Results:

DTR\_MAE: 556

Test score: -1.40

Random Forest Regressor Results Train:

Test score: 0.61

Random Forest Regressor Results:

RF\_MAE: 374

Test score: -0.10

K Neighbors Regressor Results Train:

Test score: 0.20

K Neighbors Regressor Results:

KNN\_MAE: 392

Test score: -0.26

Gradient Boosting Regressor Results Train:

Test score: 0.54

Gradient Boosting Regressor Results:

GBR\_MAE: 381

Test score: -0.12

**Сравним наши модели по метрике МАЕ**

|  | **Регрессор** | **MAE** |
| --- | --- | --- |
| **0** | RandomForest | 371.862402 |
| **1** | Linear Regression | 363.144068 |
| **2** | KNeighbors | 392.394492 |
| **3** | DecisionTree | 541.022823 |
| **4** | Gradient Boosting Regressor | 382.081859 |

Лучше всего себя показала модель линейной регрессии

Оптимизация с помощью GridSearch не улучшила показателей модели.

lr\_MAE\_ridge: 363.0528

#### **Прогноз модуля упругости при растяжении**

Linear Regression Results Train:

Test score: 0.02

Linear Regression Results:

lr\_MAE: 2

Test score: -0.01

Decision Tree Regressor Results Train:

Test score: 1.00

Decision Tree Regressor Results:

DTR\_MAE: 4

Test score: -1.16

Random Forest Regressor Results Train:

Test score: 0.43

Random Forest Regressor Results:

RF\_MAE: 2

Test score: -0.04

K Neighbors Regressor Results Train:

Test score: 0.16

K Neighbors Regressor Results:

KNN\_MAE: 3

Test score: -0.22

Gradient Boosting Regressor Results Train:

Test score: 0.46

Gradient Boosting Regressor Results:

GBR\_MAE: 2

Test score: -0.09

**Сравним наши модели по метрике МАЕ**

|  | **Регрессор** | **MAE** |
| --- | --- | --- |
| **0** | RandomForest | 2.415135 |
| **1** | Linear Regression | 2.414573 |
| **2** | KNeighbors | 2.661645 |
| **3** | DecisionTree | 3.496319 |
| **4** | Gradient Boosting Regressor | 2.493065 |

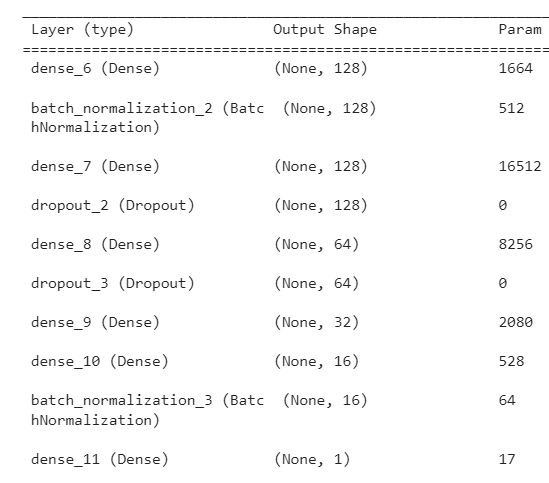
Лучше всего себя показала модель линейной регрессии

Оптимизация с помощью GridSearch не улучшила показателей модели.

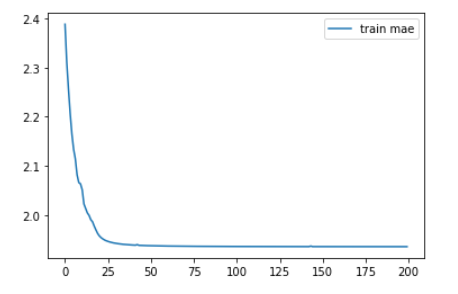
lr\_MAE\_ridge: 2.4154

* 1. *Написать нейронную сеть, которая будет рекомендовать соотношение матрица.*

Архитектура сети:



На вход подаются 128 нейронов, на выходе 1, между ними скрытые слои. Функция активация relu, оптимизатор – adam, метрика – mae.



На графике видно, что модель обучилась, после 27 эпохи ошибка падает очень медленно.

* 1. *Разработка приложения*

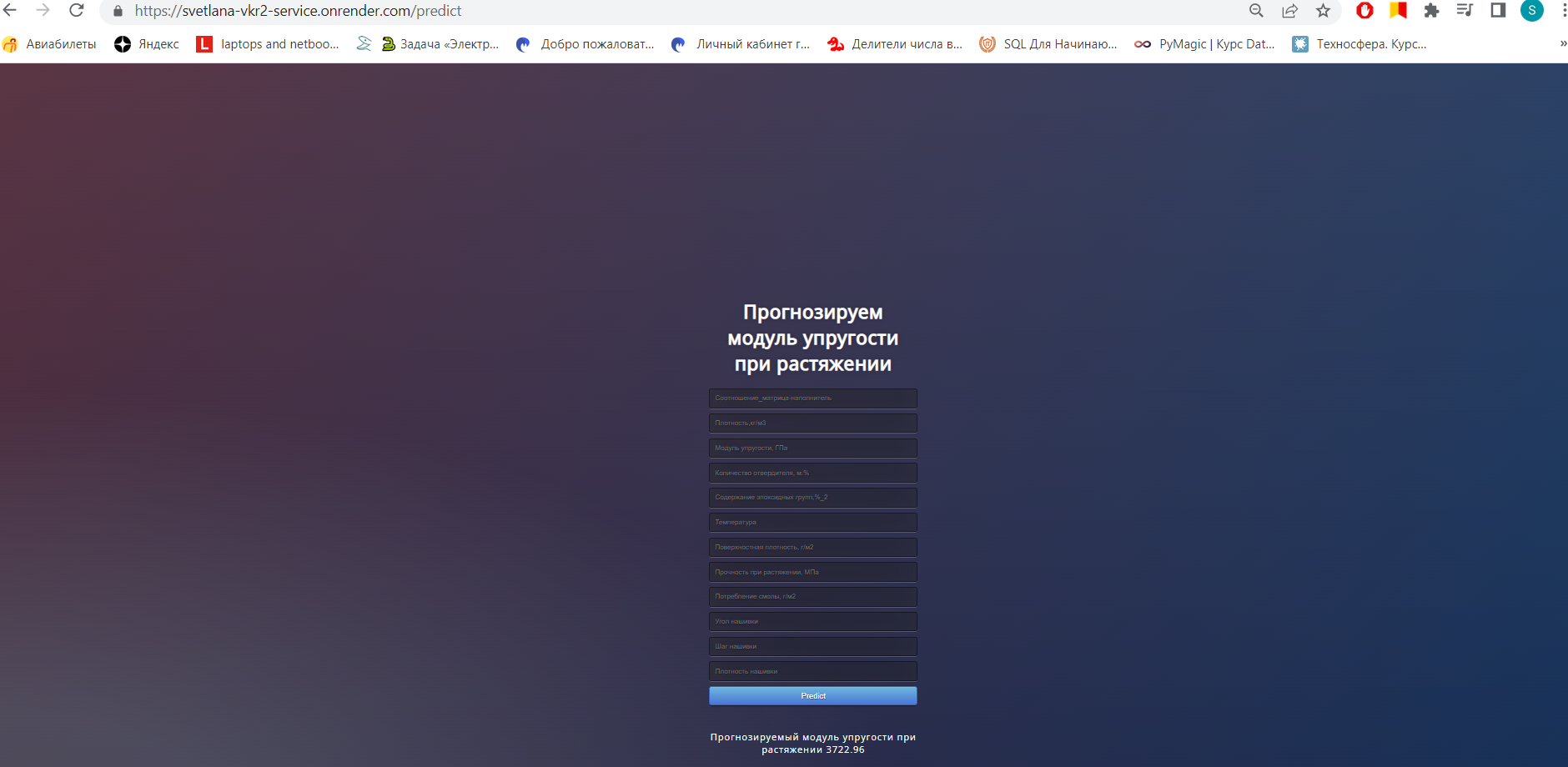
На основе модели линейно регрессии разработано приложение, которое предсказывает модуль упругости при растяжении.

Приложение создано с помощью фреймворк Flask, развернуто на render.com

Ссылка на приложение:

<https://svetlana-vkr3-service.onrender.com/predict>

Нужно ввести 12 признаков и на выходе будет выведен прогноз упругости при растяжении.



* 1. *Создание удаленного репозитория и загрузка результатов работы на него.*

Все материалы находятся в репозитории на gihub

https://github.com/SvetlanaMatsievskaya/Matsievskaya\_SG\_materials\_for\_VKR.git