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»

**Прогнозирование конечных свойств новых материалов (композитных материалов)**

Слушатель Семенова Екатерина Николаевна

Новосибирск, 2023

Содержание

[**Введение** 3](#_Toc105968923)

[**Глава 1. Аналитическая часть** 9](#_Toc105968924)

[**1.1 Постановка задачи** 9](#_Toc105968925)

[**1.2 Описание используемых методов** 10](#_Toc105968926)

[**Глава 2. Практическая часть** 13](#_Toc105968927)

[**2.1. Предобработка данных** 13](#_Toc105968928)

[**2.1 Обучение моделей** 21](#_Toc105968929)

[**2.2. Нейронная сеть для рекомендации соотношения матрица-наполнитель** 26](#_Toc105968930)

[**Заключение** 2](#_Toc105968933)7

# **Введение**

# Непрерывное развитие науки и техники, возникновение новых технологий, продвижение исследований вглубь океанов и космоса влекут за собой разработку современных материалов, позволяющих повышать эффективность и надёжность конструкций. Традиционные конструкционные материалы достигли максимальных пределов своих механических и химических характеристик. Они отступают под натиском более совершенных композиционных материалов, разработке которых придаётся огромное значение во всем мире. Улучшение свойств современных композитов — это снижение потребления энергетических ресурсов, минимизация влияния на экологию, действенный метод борьбы с коррозией, рост конструкторского и технологического потенциала.

# Композитный материал или композит— многокомпонентный материал, изготовленный (человеком или природой) из двух или более компонентов с существенно различными физическими и/или химическими свойствами, которые, в сочетании, приводят к появлению нового материала с характеристиками, отличными от характеристик отдельных компонентов и не являющимися простой их суперпозицией. В составе композита принято выделять матрицу/матрицы и наполнитель/наполнители, последние выполняют функцию армирования (как, например, арматура в железобетоне). В качестве наполнителей композитов как правило выступают углеродные или стеклянные волокна, а роль матрицы играет полимер. Сочетание разных компонентов позволяет улучшить характеристики материала и делает его одновременно лёгким и прочным. При этом отдельные компоненты остаются таковыми в структуре композитов, что отличает их от смесей и затвердевших растворов. Варьируя состав матрицы и наполнителя, их соотношение, ориентацию наполнителя, получают широкий спектр материалов с требуемым набором свойств. Многие композиты превосходят традиционные материалы и сплавы по своим механическим свойствам и в то же время они легче. Использование композитов обычно позволяет уменьшить массу конструкции при сохранении или улучшении её механических характеристик.

# В настоящее время практически ни одна из солидных международных выставок не проходит без присутствия решений для базальтокомпозитов. Базальтокомпозит – это композит, в основу которого входит базальтовое волокно. Базальтовое волокно в свою очередь является продуктом вулканической деятельности. За счет первичного плавления базальтовых пород и их гомогенизация уже произведены самой природой, а это ощутимо снижает стоимость получаемого продукта. Немаловажной характеристикой базальтового волокна является его прекрасная совместимость со всевозможными матричными материалами: полимерными, керамическими, металлическими, цементными, углеродными — и различными волокнами для получения гибридных композиционных материалов. Для изготовления базальтокомпозитов применимы практически все известные методы: намотка, формование, напыление, литьё и прессование.

# Заинтересованность такими гибридными композитами растет с каждым днем, но возникает одна «проблема»: даже если мы знаем свойства каждого материала, каким получится композит из этих материалов – сложно сказать. Для решения этого вопроса есть два пути: физические испытания образцов материалов (что является трудно затратным занятием) или прогнозирование характеристик.

# Суть прогнозирования заключается в симуляции представительного элемента объема композита, на основе данных о характеристиках входящих компонентов (связующего и армирующего компонента). Наша задача состоит как раз в этом. Необходимо изучить данные по базальтопластику, а также данные по углепластиковым нашивкам. Исходя из анализа данных, определить свойства гибридного композита. В ходе работы были разработаны и протестированы несколько моделей для свойств «Модуль упругости при растяжении» и «Прочность при растяжении», а также написана нейронная сеть, способная определить соотношение «матрица – наполнитель». На основе нейронной сети было написано пользовательское приложение. Все материалы можно найти на GitHub (ссылку на репозиторий можно найти в разделе 2.6 GitHub на странице 30)

# **Глава 1. Аналитическая часть**

## **1.1 Постановка задачи**

Необходимо спрогнозировать ряд конечных свойств получаемых композиционных материалов.

В их перечень входит:

* Модуль упругости при растяжении, Гпа
* Прочность при растяжении, Мпа

Анализ необходимо проводить на основе следующих свойств:

* 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

Задача основан на реальных производственных задачах Центра НТИ «Цифровое материаловедение: новые материалы и вещества» (структурное подразделение МГТУ им. Н.Э. Баумана).

Заинтересованными лицами могут выступать Российские металлурги:

* ГМК НорНикель
* Северсталь
* Русал
* En+
* НЛМК
* Росатом
* И т.д.

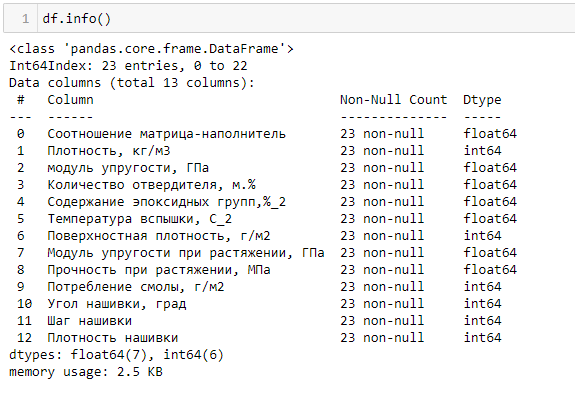
Созданные прогнозные модели помогут сократить количество проводимых испытаний, а также пополнить базу данных материалов возможными новыми характеристиками материалов, и цифровыми двойниками новых композитов.

На входе есть два датасета со своими признаками (свойствами). Первый набор данных data\_bp имеет в себе 1023 значения, 10 признаков и один столбец с нумерацией. Второй набор данных data\_nup имеет 1040 значений, 3 признака и один столбец с нумерацией. Для того, чтобы получить общую сводную таблицу, объединяем два датасета по inner, убираем «синтетические данные» и получаем на входе один дата сет.

В результате получаем набор данных с 1000 наблюдениями, 13 признаками и одним столбцом нумерации. Тип данных во всех колонках float64 (число с плавающей точкой), кроме столбца «Угол нашивки» - здесь у нас тип данных int (целочисленное значение). В данном наборе наблюдаются наши целевые признаки: модуль упругости при растяжении, прочность при растяжении и соотношение матрица-наполнитель. Так как нам есть на что опираться, можно сделать вывод, что это будет обучение с учителем.

Прежде чем преступить к решению нашей задачи, необходимо рассмотреть данные подробнее: проверить на пропущенные значения, дубликаты, статистические показатели (mean, std, median).

****



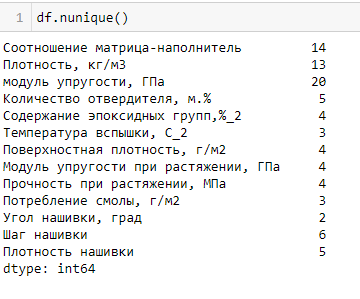


Рисунок 1, 2, 3 – статистические показатели по всем признакам датасета

Также необходимо нарисовать гистограммы распределения каждой из переменной, диаграммы ящика с усами, попарные графики рассеяния точек.

После небольшого анализа можно описать наши данные: пропусков и дубликатов не обнаружено, но почти в каждом признаке видны выбросы. В дальнейшем выбросы нужно будет удалить, так как есть модели машинного обучения, которые очень «чувствительны» к выбросам. Если присмотреться к статистическим показателям, то можно увидеть и разнородность данных.

Подобное можно увидеть на графике».

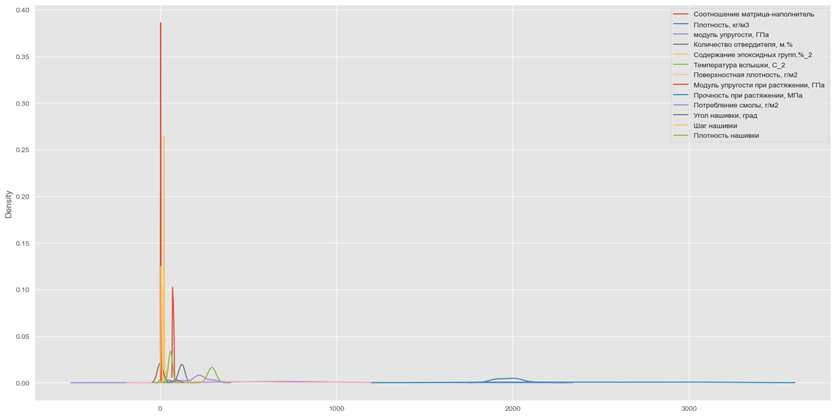


Рисунок 4 - оценка плотности ядра до очистки данных

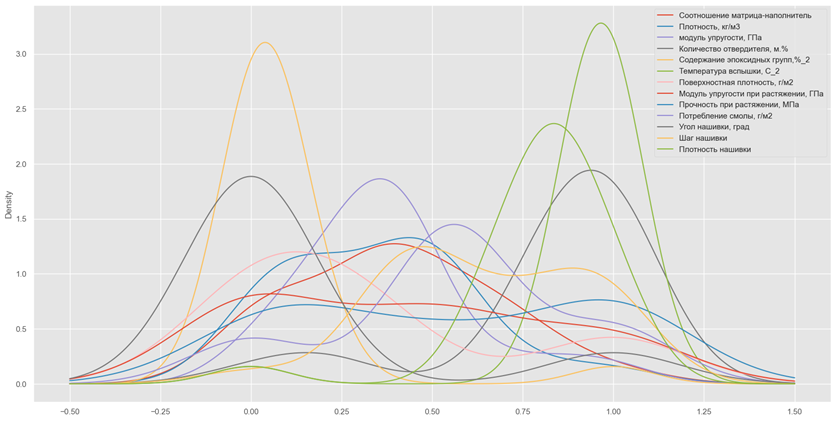


Рисунок 5 - оценка плотности ядра после очистки данных

Так как у нас решается задача регрессии, то нам важны взаимосвязи между признаками. Поэтому посмотрим на корреляционную матрицу датасета по всем признакам:

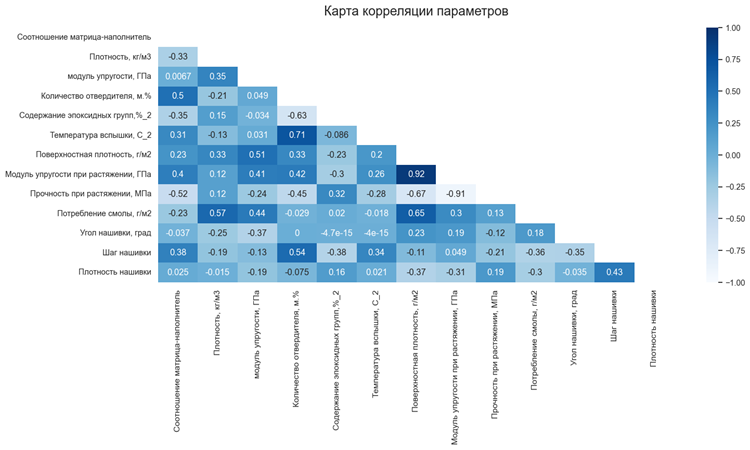


Рисунок 6 – корреляционная матрица для композитных данных.

Корреляция - важнейший фактор, лежащий в основе анализа данных. Он сообщает, как переменные в наборе данных связаны друг с другом и как они перемещаются относительно друг друга. Значение корреляции колеблется от -1 до +1. 0 Корреляция означает, что две переменные не зависят друг от друга. Положительная корреляция указывает на то, что переменные движутся в одном направлении, а отрицательная корреляция указывает на противоположное.

Мы можем построить корреляционную матрицу с помощью модуля seaborn. Это помогает легко понять набор данных и очень часто используется для анализа.

По графику видно, что наблюдается коррелляция между упругостью при растяжении и поверхностной плотностью.

Я не знаю, почему так получилось, потому что ничего не понимаю в физике материалов.

Теперь можно сказать, какой пласт работы нас ждет: необходимо очистить датасет от выбросов, провести нормализацию и стандартизацию данных, после чего посмотреть, появятся ли какие-либо связи между признаками.

## **1.2 Описание используемых методов**

Для прогнозирования модуля упругости при растяжении и прочности при растяжении применим несколько методов машинного обучения.

* Случайный лес решений
* Логистическая регрессия

**Метод «Случайный лес»**

Алгоритм машинного обучения, заключающийся в использовании комитета (ансамбля) решающих деревьев. Алгоритм сочетает в себе две основные идеи: метод бэггинга Бреймана, и метод случайных подпространств, предложенный Тин Кам Хо. Алгоритм применяется для задач классификации, регрессии и кластеризации. Основная идея заключается в использовании большого ансамбля решающих деревьев, каждое из которых само по себе даёт очень невысокое качество классификации, но за счёт их большого количества результат получается хорошим.

Сама структура дерева представляет собой «листья» и «ветки». На рёбрах («ветках») дерева решения записаны признаки, от которых зависит целевая функция, в «листьях» записаны значения целевой функции, а в остальных узлах — признаки, по которым различаются случаи. Чтобы классифицировать новый случай, надо спуститься по дереву до листа и выдать соответствующее значение.

Подобные деревья решений широко используются в интеллектуальном анализе данных. Цель состоит в том, чтобы создать модель, которая предсказывает значение целевой переменной на основе нескольких переменных на входе.

Каждый лист представляет собой значение целевой переменной, изменённой в ходе движения от корня по рёбрам дерева до листа. Каждый внутренний узел сопоставляется с одной из входных переменных. Дерево может быть также «изучено» разделением исходных наборов переменных на подмножества, основанные на проверке значений признаков. Это действие повторяется на каждом из полученных подмножеств. Рекурсия завершается тогда, когда подмножество в узле имеет те же значения целевой переменной, таким образом, оно не добавляет ценности для предсказаний.

**Линейная регрессия**

Используемая в статистике регрессионная модель зависимости одной (объясняемой, зависимой) переменной от другой или нескольких других переменных (факторов, регрессоров, независимых переменных) с линейной функцией зависимости. Модель линейной регрессии является часто используемой и наиболее изученной в эконометрике. А именно изучены свойства оценок параметров, получаемых различными методами при предположениях о вероятностных характеристиках факторов, и случайных ошибок модели. Предельные (асимптотические) свойства оценок нелинейных моделей также выводятся исходя из аппроксимации последних линейными моделями. С эконометрической точки зрения более важное значение имеет линейность по параметрам, чем линейность по факторам модели.

Регрессионная модель



где b — параметры модели,  — случайная ошибка модели; называется линейной регрессией, если функция регрессии f(x,b) имеет вид



где  — параметры (коэффициенты) регрессии,  — регрессоры (факторы модели), k — количество факторов модели. Коэффициенты линейной регрессии показывают скорость изменения зависимой переменной по данному фактору, при фиксированных остальных факторах (в линейной модели эта скорость постоянна).

**1.3 Разведочный анализ**

В данном разделе приведем краткое описание методов разведочного анализа данных, которые используются для первоначального анализа.

Чтобы получить всю информацию о датасете в одном месте, можно воспользоваться pandas\_profiling. Применяя метод profile\_report() к датасету, получаем разбор всех признаков, по ним видим пропущенные значения, графики распределения признаков, описательную статистику, корреляцию и много другого.

# 

# **Глава 2. Практическая часть**

## **2.1. Предобработка данных**

Загружаем 2 файла дата сета

data\_bp = pd.read\_excel('content/X\_bp.xlsx', index\_col=0)

data\_nup = pd.read\_excel('content/X\_nup.xlsx', index\_col=0)

Объединяем выше указанные файлы:

df = data\_bp.merge(data\_nup, left\_index=True, right\_index=True, how='inner')

Выводим первые 5 строк на рисунке 1



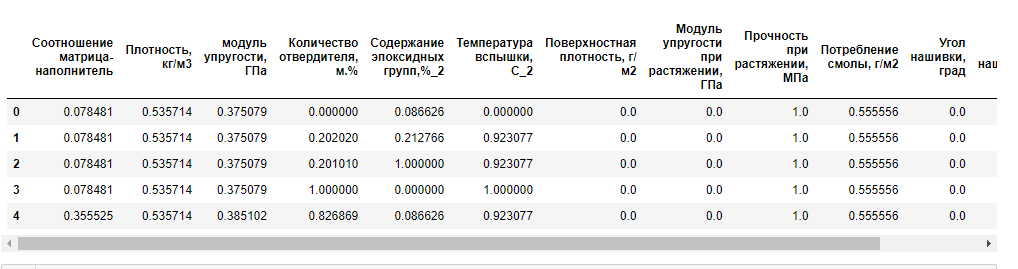


Рисунок 1. Первые 4 строки объединенного датасета

Выводим описательную статистику по датасету на рисунке 2

df.describe().round(2).T

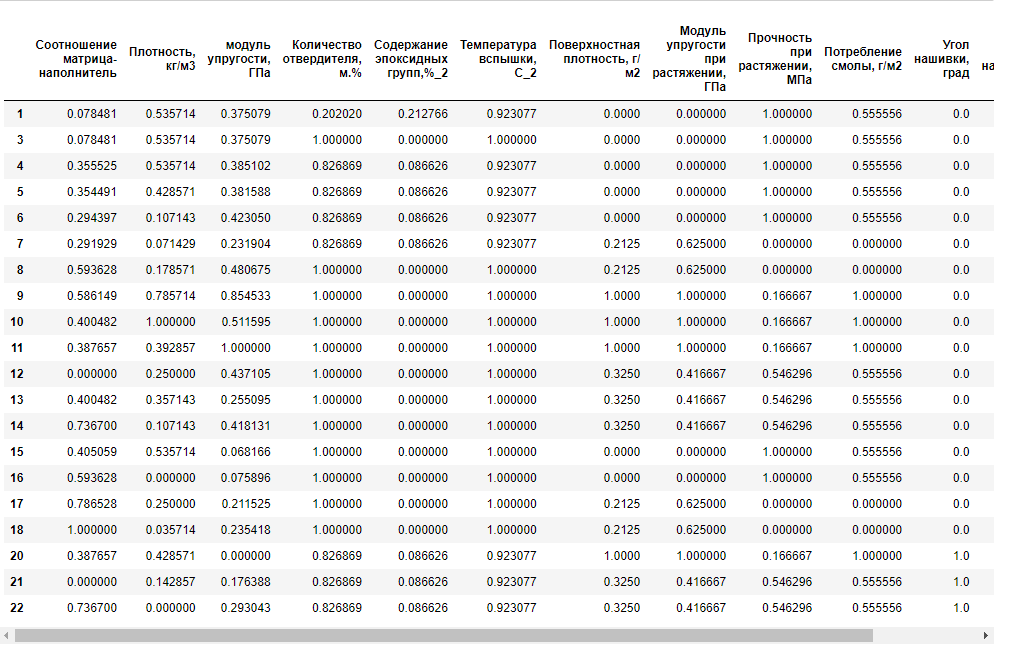


Рисунок 2. Таблица описательной статистики по объединенному датасету

Проверяем:

* Корреляцию
* Плотность распределение

sns.set(style = 'white', palette ='dark')

grids = sns.PairGrid(df, diag\_sharey = False, height =4)

grids.map\_diag(sns.distplot, kde = True)

grids.map\_upper(sns.kdeplot)

grids.map\_lower(sns.scatterplot)

Смотрим графики корреляции и плотности распределения на рисунке 3

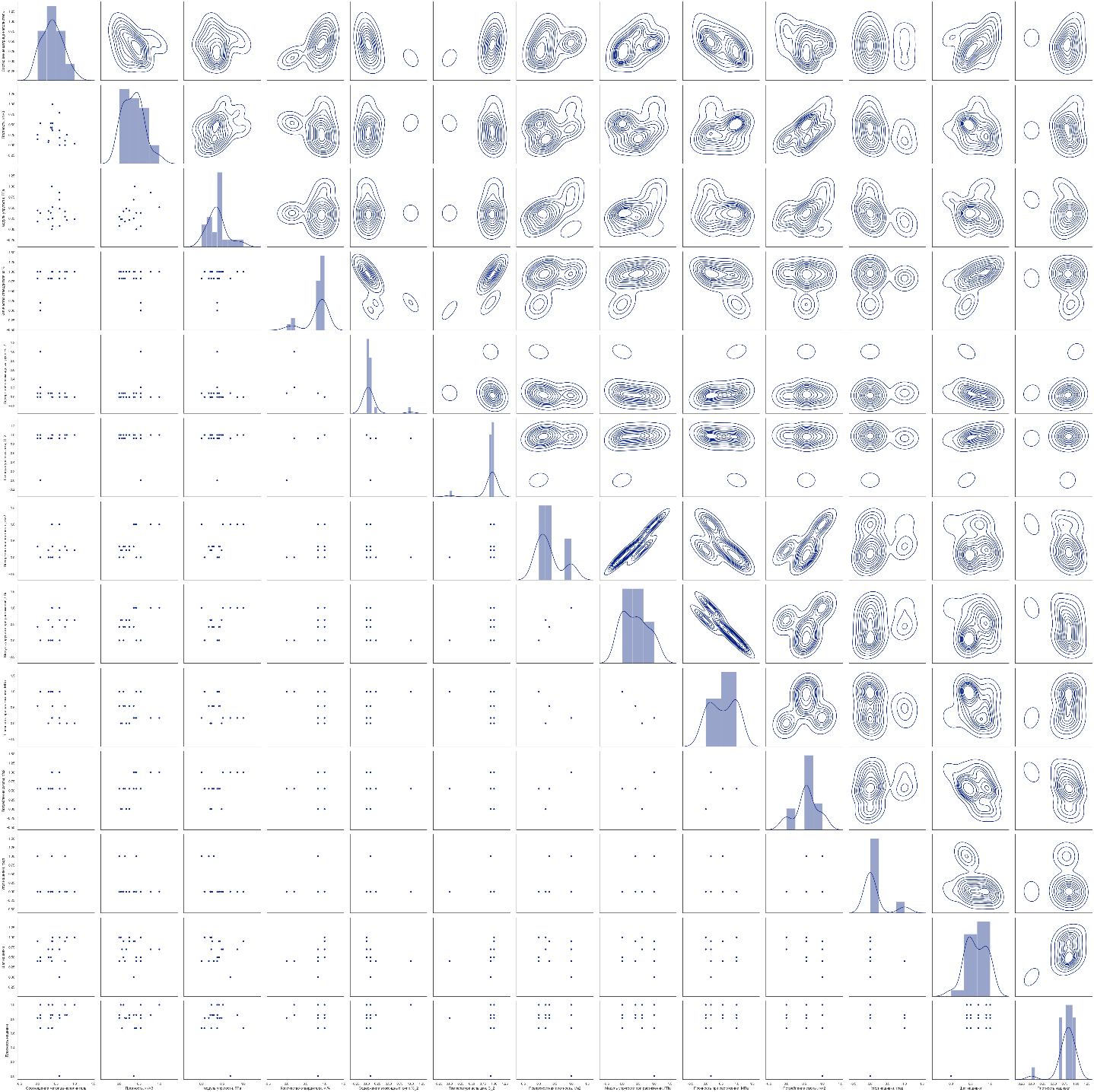


Рисунок 3. График корреляции и плотности распределния

Распределение приближено к нормальному. Отсутствует корреляция

Дополнительно строим тепловую матрицу:

plt.figure(figsize=(16, 6))

mask = np.triu(np.ones\_like(df.corr(), dtype=np.bool))

heatmap = sns.heatmap(df.corr(), mask=mask, vmin=-1, vmax=1, annot=True, cmap='Blues')

heatmap.set\_title('Карта корреляции параметров', fontdict={'fontsize':18}, pad=16)

Проверим на выбросы. Построим график ящик с усами, который показывает выбросы на рисунке 4.

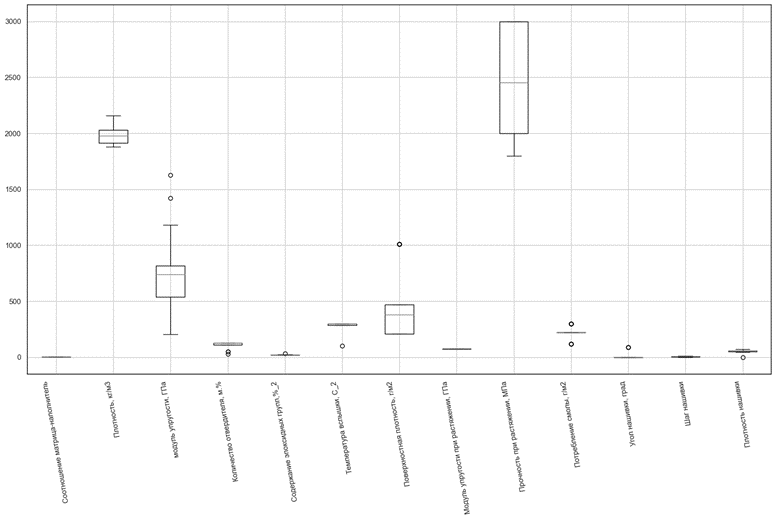


Рисунок 4 – «ящик с усами» до нормализации дата сета.

Для того, чтобы график был информативным, предварительно надо нормализовать датасет.

Убираем столбцы и нормализуем датасет

minmax\_scaler = MinMaxScaler()

df\_normal = minmax\_scaler.fit\_transform(np.array(df\_normal[['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']]))

Возвращаем столбцы

dataset\_norm\_df = pd.DataFrame(data = df\_normal, columns = ['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'])

dataset\_norm\_df.head()

Смотрим график ящик с усами, который показывает выбросы на рисунке 5

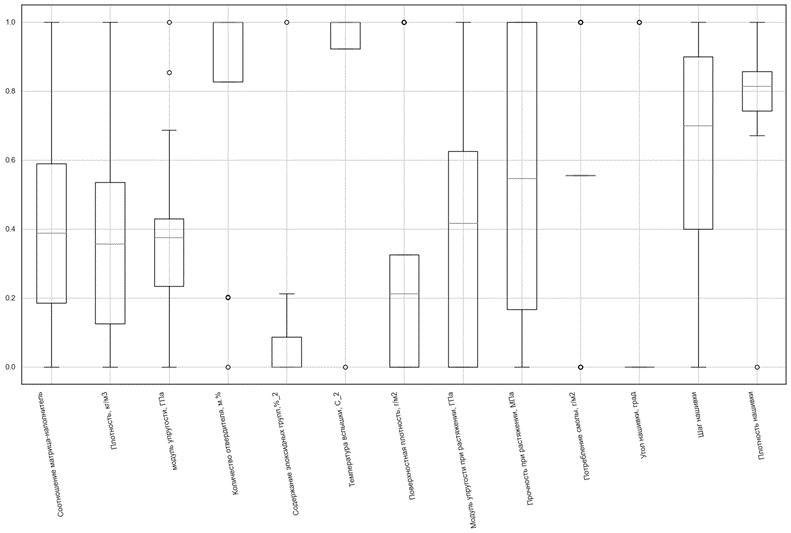


Рисунок 5. Ящик с усами

Убираем выбросы по правилу 3 сигм. После удаления выбросов в датасете остается 22 записи.

sigma\_normal = dataset\_norm\_df[(np.abs(stats.zscore(dataset\_norm\_df)) < 3).all(axis=1)]

sigma\_normal

Проведем прямую через точки для «Модуль упругости при растяжении, ГПа»

Для подтверждения метода регрессии для искомых свойств композитов попробуем провести прямую через точки

sns.pairplot(df\_MinMax, x\_vars=["Соотношение матрица-наполнитель", "Прочность при растяжении, МПа", 'Плотность, кг/м3',

                                'модуль упругости, ГПа', 'Количество отвердителя, м.%', 'Содержание эпоксидных групп,%\_2',

                                'Температура вспышки, С\_2', 'Поверхностная плотность, г/м2', 'Потребление смолы, г/м2',

                                'Угол нашивки, град', 'Шаг нашивки', 'Плотность нашивки'], y\_vars='Модуль упругости при растяжении, ГПа', size=7, aspect=0.8,kind = 'reg')

plt.savefig("pairplot.jpg")

plt.show()

Смотрим на график регресии для Модуля упругости при растяжении, Гпа на рисунке 6

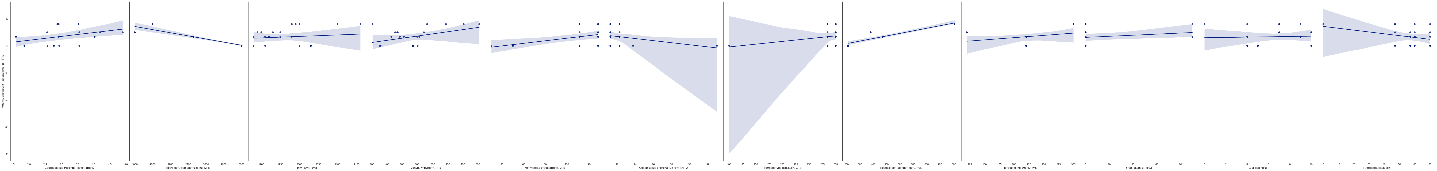


Рисунок 6. График регрессии Модуля упругости при растяжении, ГПа

Проведем прямую через точки для «Прочность при растяжении, МПа»

По аналогии строим график для второго искомого свойства

sns.pairplot(dataset\_norm\_df, x\_vars=["Соотношение матрица-наполнитель", "Модуль упругости при растяжении, ГПа", 'Плотность, кг/м3',

                                'модуль упругости, ГПа', 'Количество отвердителя, м.%', 'Содержание эпоксидных групп,%\_2',

                                'Температура вспышки, С\_2', 'Поверхностная плотность, г/м2', 'Потребление смолы, г/м2',

                                'Угол нашивки, град', 'Шаг нашивки', 'Плотность нашивки'], y\_vars='Прочность при растяжении, МПа', size=10, aspect=0.7,kind = 'reg')

plt.show()

Смотрим график регрессии для свойства Прочность при растяжении, Мпа на рисунке 7

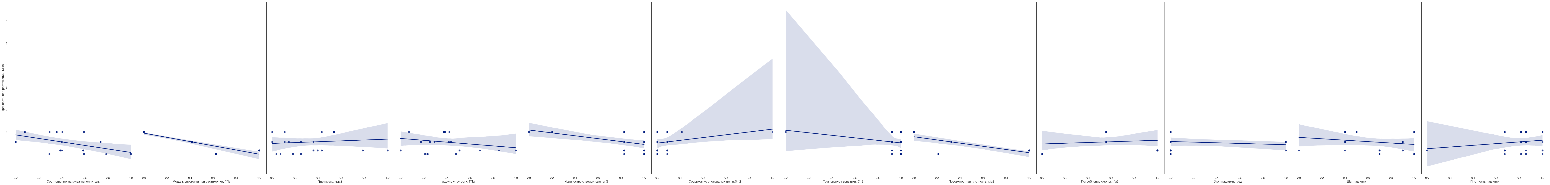


Рисунок 7. График регрессии Прочность при растяжении, Мпа

## **2.2 Обучение моделей**

Создаем переменные для наших тестовых и обучающих выборок.

Не нормализованный датасет:

* х\y\_1 - Модуль упругости при растяжении, ГПа
* х\y\_2 - Прочность при растяжении, МПа
* х\y\_all - объединенный

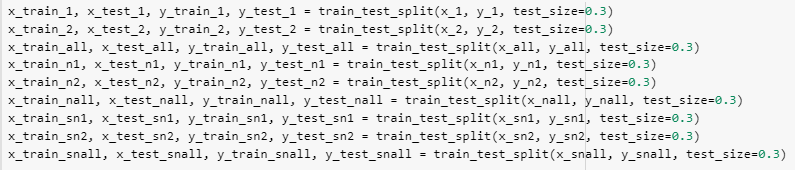
Нормализованный датасет:

* х\y\_n1 - Модуль упругости при растяжении, ГПа
* х\y\_n2 - Прочность при растяжении, МПа
* х\y\_nall – объединенный

Нормализованный датасет без выбросов:

* х\y \_sn1 - Модуль упругости при растяжении, ГПа
* х\y \_sn2 - Прочность при растяжении, МПа
* х\y \_snall – объединенный

Разбиваем датасет на тестовую и обучающую выборку



Случайный лес:

Для среднеквадратичной и абсолютной ошибки будем использовать параметры:



Для score будем использовать лучшее дерево решений

print(best\_dec\_tree\_1)

print(best\_dec\_tree\_2)

print(best\_dec\_tree\_all)

print(best\_dec\_tree\_n1)

print(best\_dec\_tree\_n2)

print(best\_dec\_tree\_nall)

print(best\_dec\_tree\_sn1)

print(best\_dec\_tree\_sn2)

print(best\_dec\_tree\_snall)

RandomForestRegressor(ccp\_alpha=0.01, max\_depth=3, max\_features='sqrt')

RandomForestRegressor(ccp\_alpha=0.01, max\_depth=3, max\_features='log2')

RandomForestRegressor(ccp\_alpha=0.1, max\_depth=2, max\_features='log2')

RandomForestRegressor(ccp\_alpha=0.001, max\_depth=4, max\_features='log2')

RandomForestRegressor(ccp\_alpha=0.001, max\_depth=2)

RandomForestRegressor(ccp\_alpha=0.001, max\_depth=3)

RandomForestRegressor(ccp\_alpha=0.001, max\_depth=3)

RandomForestRegressor(ccp\_alpha=0.001, max\_depth=3)

RandomForestRegressor(ccp\_alpha=0.001, max\_depth=4)

Выводим результаты предсказания:

1

1

R2: 0.9287688433734942

MSE: 0.6703158730158723

MAE: 0.6823809523809489

2

R2: 0.9368047699001611

MSE: 17638.299823633013

MAE: 109.07936507936415

all

R2: 0.9252106960219455

MSE: 18434.319034479686

MAE: 59.07103174603164

n1

R2: 0.9363738744320522

MSE: 0.010915451388888992

MAE: 0.07172619047619058

n2

R2: 0.9726358742785497

MSE: 0.004076638827088073

MAE: 0.04103826530612293

nall

R2: 0.9634179128543918

MSE: 0.0060073182798925625

MAE: 0.048815586419752845

sn1

R2: 0.982218613861386

MSE: 0.0021652199074074312

MAE: 0.03125000000000051

sn2

R2: 0.8317876210360708

MSE: 0.019858405849908296

MAE: 0.13354166666666584

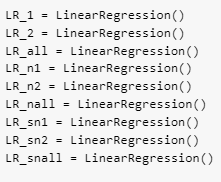
snall

R2: 0.9665970868445812

MSE: 0.0025595225280117297

MAE: 0.04062453152557221

Линейная регрессия



Выводим результат предсказания:

LR1

R2: 1.0

MSE: 2.8849770248082716e-29

MAE: 2.0301221021717147e-15

LR2

R2: 1.0

MSE: 7.194994220974639e-23

MAE: 4.612437416134136e-12

LRall

R2: 1.0

MSE: 9.65392447420638e-22

MAE: 1.015670087716509e-11

LRn1

R2: 1.0

MSE: 4.059860322768293e-31

MAE: 5.114956077737328e-16

LRn2

R2: 1.0

MSE: 1.3527731929375945e-30

MAE: 8.48527597392084e-16

LRnall

R2: 0.9999993926336272

MSE: 8.630515593883189e-08

MAE: 0.00020685907322122907

LRsn1

R2: 1.0

MSE: 3.502624592192253e-31

MAE: 4.996003610813204e-16

LRsn2

R2: 1.0

MSE: 2.475461955185727e-31

MAE: 4.070817756958907e-16

LRsnall

R2: 0.9998472454968176

MSE: 1.1230057549214057e-05

MAE: 0.0013680863685103287

Надо заниматься тем, что понятно, например, транспортной логистикой или юриспруденцией.

**2.2.** **Нейронная сеть для рекомендации соотношения матрица-наполнитель**

Обучаем переменные X и У:



Создаем структуру многослойный персептрон. Используем 4 слоя Dense.

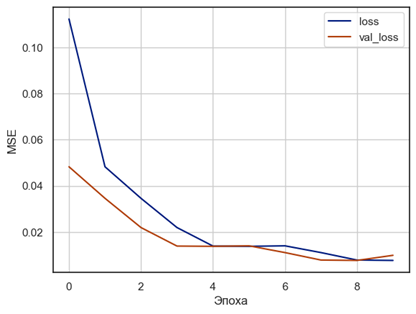
Входной слой Dense. Задаем 128 нейронов и активационную функцию relu

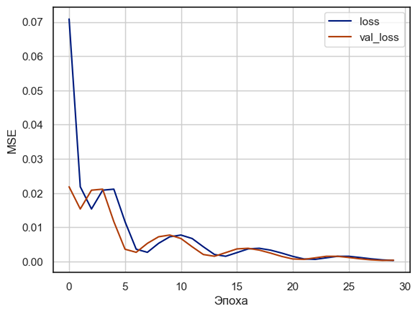
Скрытый слой Dense. Задаем 64 нейронов и активационную функцию relu

Скрытый слой Dense. Задаем 32 нейронов и активационную функцию linear

Выходной слой Dense c 1 нейроном.

Я задала 10, 30 и 100 эпох обучения для того, чтобы понять, изменятся ли результаты уменьшения ошибок.





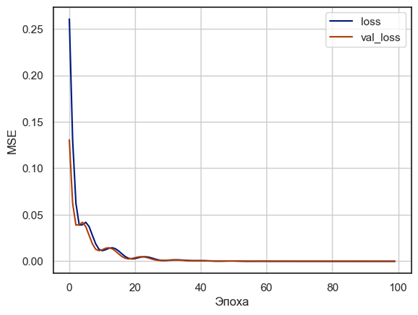


Рисунок 8 - 10 – графики уменьшения ошибок.

Выводим таблицу уменьшения ошибок:

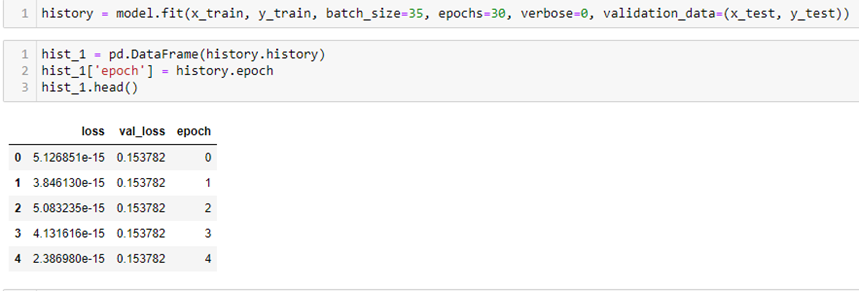


Рисунок 11 – Таблица уменьшения ошибок.

Лучше всего сработала модель 4 при 30 эпохах, потому что по графикам 30 эпох является оптимальным обучением в данном случае.

## **2.4. Создание удаленного репозитория и загрузка результатов работы на него**

Репозиторий с данными по проекту размещен на github.com

Ссылка: https://github.com/EkaterinaAisberg/-

# **Заключение**

При анализе входных данных на различных моделях машинного обучения можно сделать вывод, что для решения данной задачи не хватает данных. Ни одна из моделей не показала устойчивого хорошего результата.

# **Библиографический список**

1. <http://www.detalmach.ru/composit.htm> - Введение в композиционные материалы
2. <https://ru.wikipedia.org/wiki/Random_forest> - Случайный лес
3. <https://ru.wikipedia.org/wiki/Линейная_регрессия> - Линейная регрессия
4. А.Г.Гужва, С.А.Доленко, И.Г.Персианцев. Методика отбора существенных входных признаков при нейросетевом решении задач регрессии. Нейрокомпьютеры: разработка, применение, 2010, №3, с.20-32.
5. В.А.Головко. От многослойных персептронов к нейронным сетям глубокого доверия: парадигмы обучения и применение. В сб.: Нейроинформатика-2015. XVII Всероссийская научно-техническая конференция с международным участием. Лекции по нейроинформатике, с.47-84. НИЯУ МИФИ, 2015