Образовательный центр МГТУ им. Н.Э. Баумана

Выпускная квалификационная работа по курсу "Data Science"

Слушатель: Ефремов Ярослав

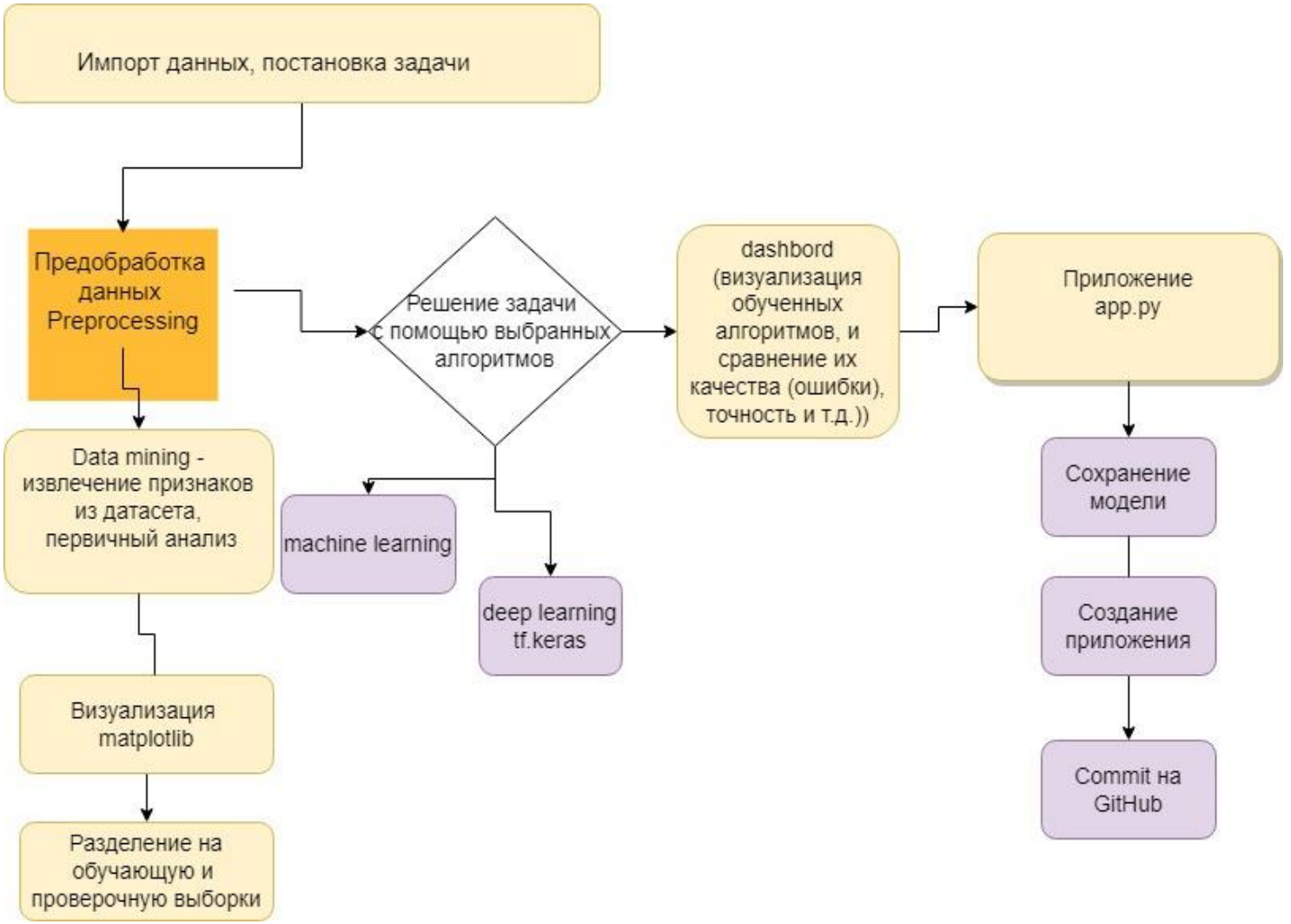
Выпускник МГТУ им. Н.Э. Баумана, работаю в отделе ADAS
(Система безопасности помощи водителю)

Тема: Прогнозирование конечных свойств новых материалов (композиционных материалов)

Постановка задачи

- Изучить предметную область
- Провести разведочный анализ данных
- Разделить данные на тренировочную и тестовую выборки
- Выполнить препроцессинг (предобаботку)
- Выбрать модель машинного обучения
- Подобрать гиперпараметры с помощью с помощью поиска по сетке с перекрестной проверкой (GridSearchCV)
- Сравнить модели после подбора гиперпараметров и выбрать лучшую модель
- Разработать нейронную сеть для с целевой переменной «Соотношение матрица-наполнитель»
- Сравнить качество лучшей модели на тренировочной и тестовой выборке
- Разработать чат-бота

Блок-схема


Импорт данных, постановка задачи
Предобработка данных Preprocessing
Решение задачи с помощью выбранных алгоритмов
dashbord (визуализация обученных алгоритмов, и сравнение их качества (ошибки), точность и т.д.))
Приложение app.py
Data mining - извлечение признаков из датасета, первичный анализ
machine learning
deep learning tf.keras
Сохранение модели
Создание приложения
Визуализация matplotlib
Commit на GitHub
Разделение на обучающую и проверочную выборки

Разведочный анализ данных

X_bp (матрица из базальтопластика):

- признаков: 10 и индекс
- строк: 1023

X_nup (наполнитель из углепластика):

- признаков: 3 и индекс
- строк: 1040

Объединение с типом INNER по индексу, получилось:

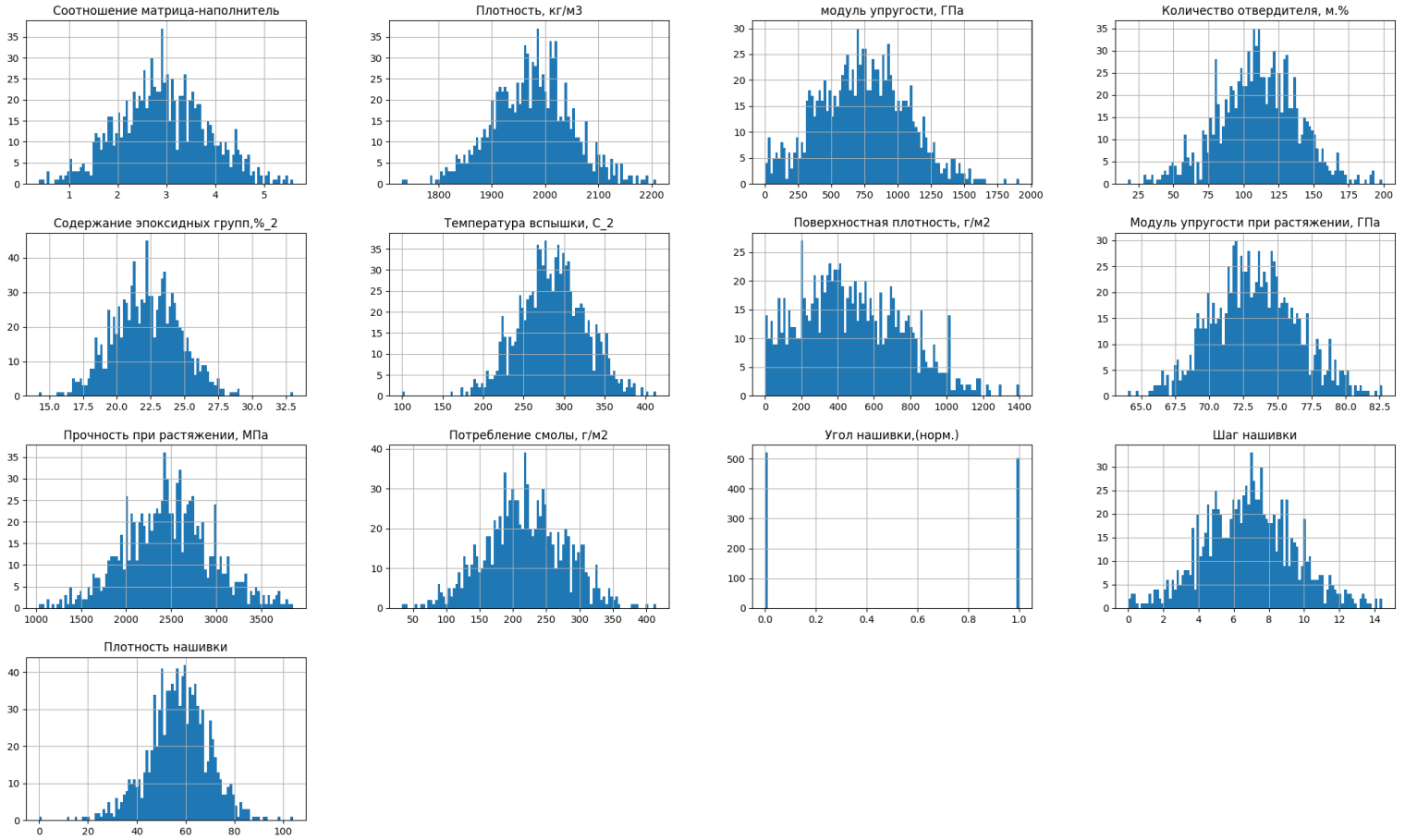
- признаков: 13
- строк: 1023

Разведочный анализ данных

Название	Файл	Тип данных	Непустых значений	Уникальных значений
Соотношение матрица-наполнител	X_bp	float64	1023	1014
Плотность, кг/м3	X_bp	float64	1023	1013
модуль упругости, ГПа	X_bp	float64	1023	1020
Количество отвердителя, м.%	X_bp	float64	1023	1005
Содержание эпоксидных групп,%_2	X_bp	float64	1023	1004
Температура вспышки, С_2	X_bp	float64	1023	1003
Поверхностная плотность, г/м2	X_bp	float64	1023	1004
Модуль упругости при растяжении, ГПа	X_bp	float64	1023	1004
Прочность при растяжении, МПа	X_bp	float64	1023	1004
Потребление смолы, г/м2	X_bp	float64	1023	1003
Угол нашивки, град	X_nup	float64	1023	2
Шаг нашивки	X_nup	float64	1023	989
Плотность нашивки	X_nup	float64	1023	988

	Среднее	Стандартное отклонение	Минимум	Максимум	Медиана
Соотношение матрица-наполнитель	2.9304	0.9132	0.3894	5.5917	2.9069
Плотность, кг/м3	1975.7349	73.7292	1731.7646	2207.7735	1977.6217
модуль упругости, ГПа	739.9232	330.2316	2.4369	1911.5365	739.6643
Количество отвердителя, м.%	110.5708	28.2959	17.7403	198.9532	110.5648
Содержание эпоксидных групп, %_2	22.2444	2.4063	14.2550	33.0000	22.2307
Температура вспышки, С_2	285.8822	40.9433	100.0000	413.2734	285.8968
Поверхностная плотность, г/м2	482.7318	281.3147	0.6037	1399.5424	451.8644
Модуль упругости при растяжении, ГПа	73.3286	3.1190	64.0541	82.6821	73.2688
Прочность при растяжении, МПа	2466.9228	485.6280	1036.8566	3848.4367	2459.5245
Потребление смолы, г/м2	218.4231	59.7359	33.8030	414.5906	219.1989
Угол нашивки, град	44.2522	45.0158	0.0000	90.0000	0.0000
Шаг нашивки	6.8992	2.5635	0.0000	14.4405	6.9161
Плотность нашивки	57.1539	12.3510	0.0000	103.9889	57.3419

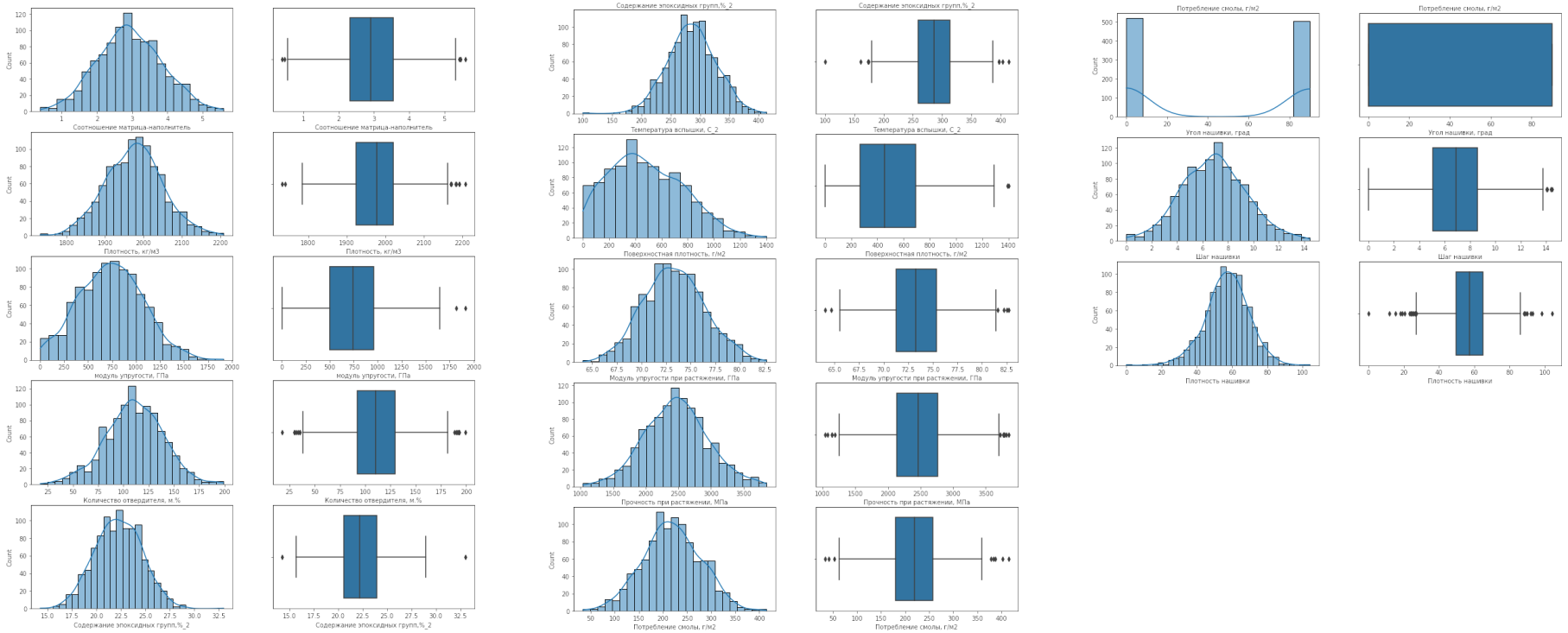
Гистограммы распределения



- Не все признаки попадают под нормального распределения

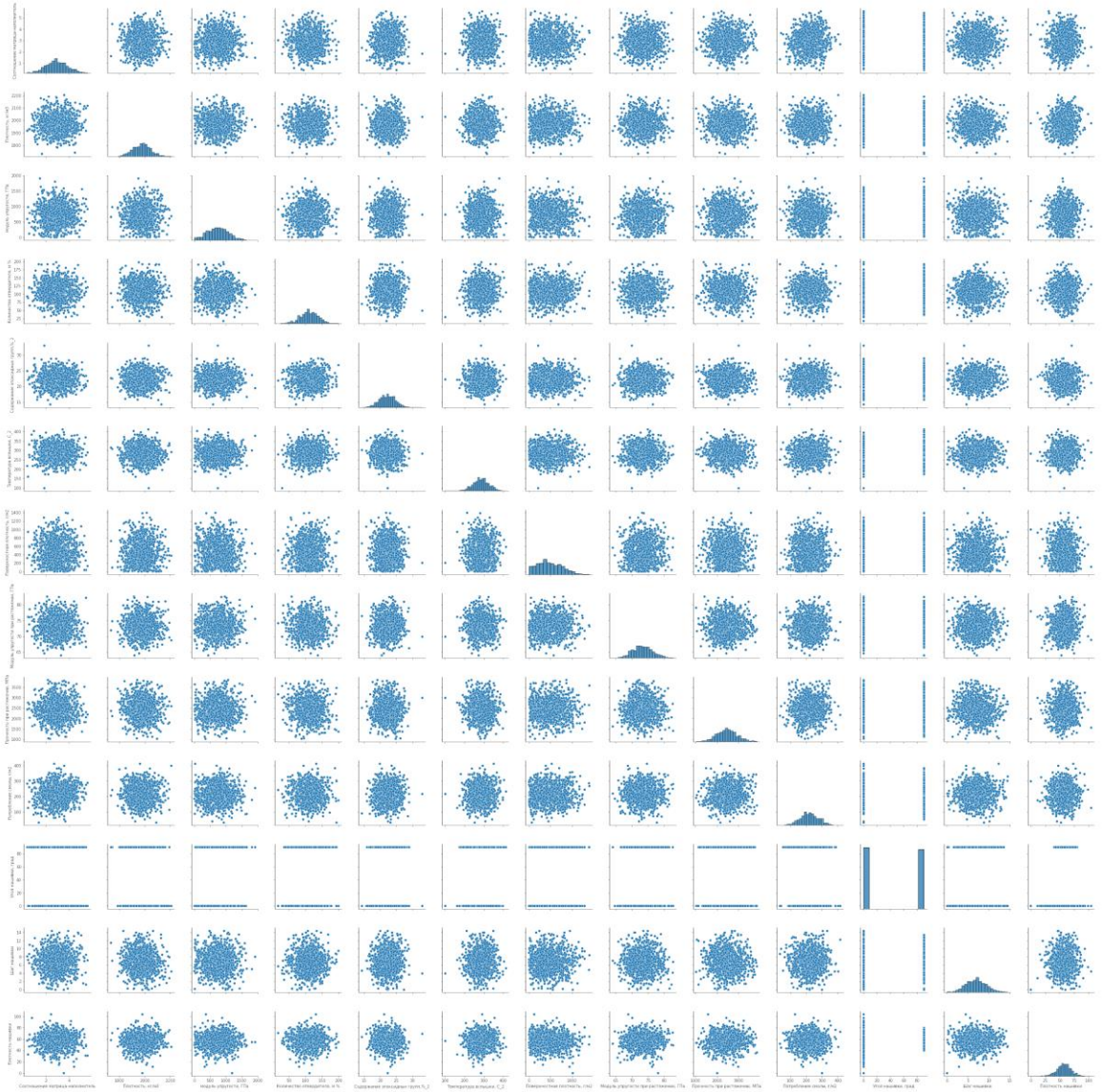
- С нормальным распределением: ['Соотношение матрица-наполнитель', 'Плотность, кг/м3', 'модуль упругости, ГПа', 'Количество отвердителя, м.%', 'Содержание эпоксидных групп,%_2', 'Температура вспышки, С_2', 'Модуль упругости при растяжении, ГПа', 'Прочность при растяжении, МПа', 'Потребление смолы, г/м2', 'Шаг нашивки']
- БЕЗ нормального распределения : ['Поверхностная плотность, г/м2', 'Угол нашивки,(норм.)', 'Плотность нашивки']

Гистограммы распределения и диаграммы “ящик с усами”



- Угол нашивки — категориальный бинарный признак
- Все остальные признаки являются количественными

Попарные графики рассеяния точек

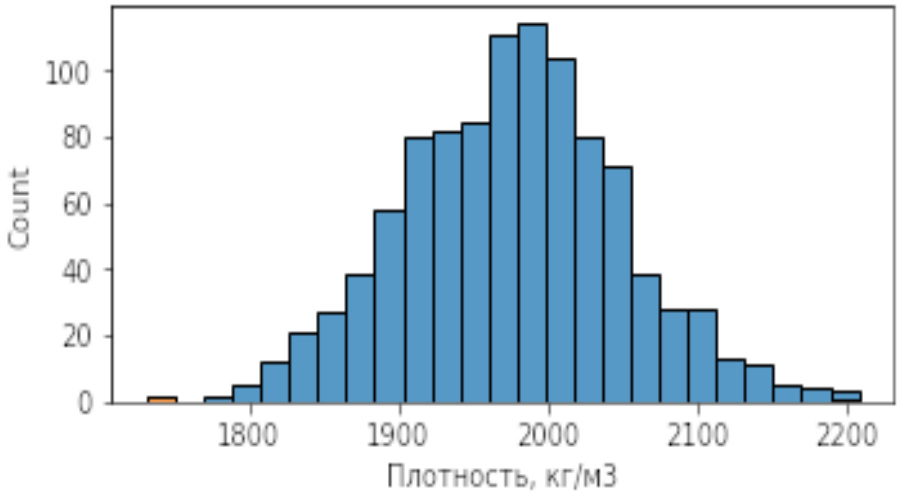


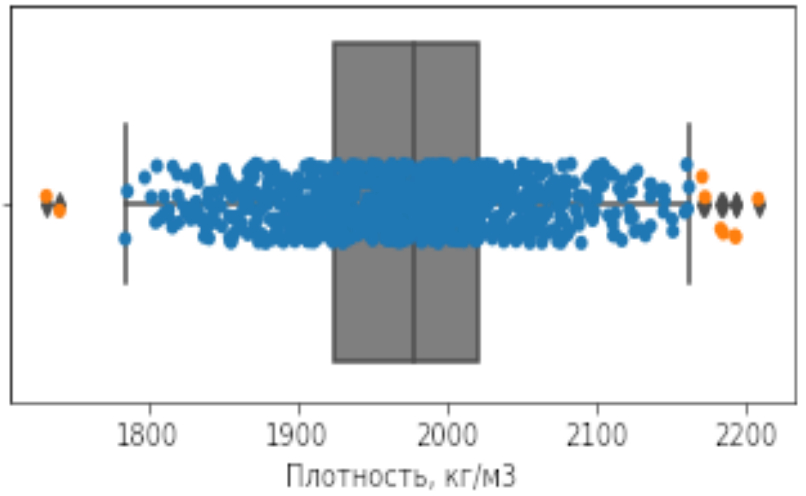
- Выбросы есть
- Не видно группировка на класстеры

Выбросы

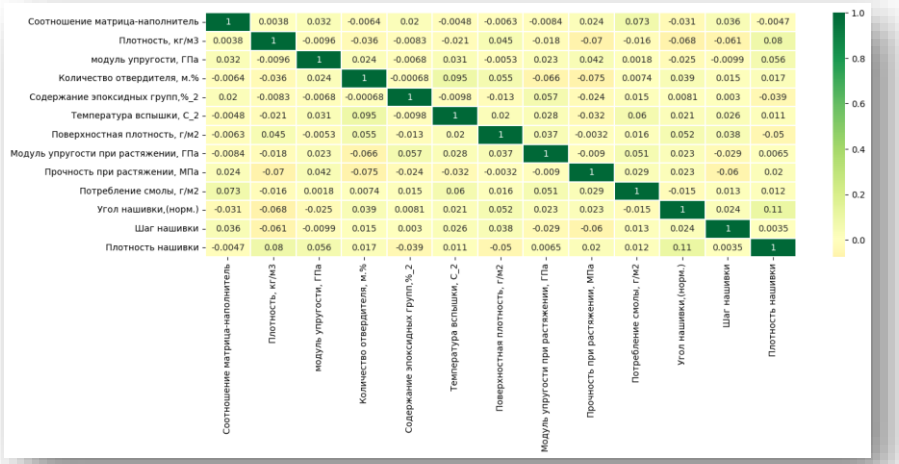
Найдено:

- методом 3-х сигм — 24 выброса
- методом межквартильных расстояний — 93 выброса
- После удаления осталось 1000 строк



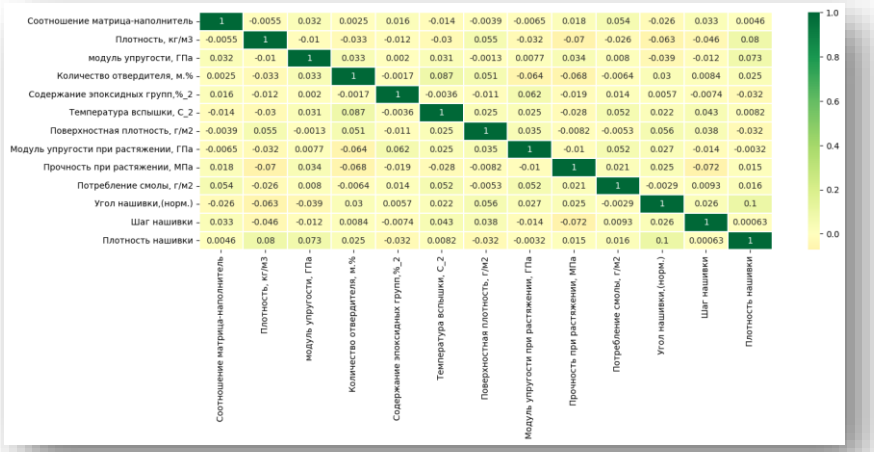


Матрица корреляции



Корреляция по Пирсону

Корреляция по Спирману



Корреляция по Кендалу

Предварительный вывод: Линейной зависимости нет!!!!!!

Предпроцессинг

Так как значения не коррелируются с друг другом, то нужно исследовать предобработку данных и получить желанную корреляцию между данными.

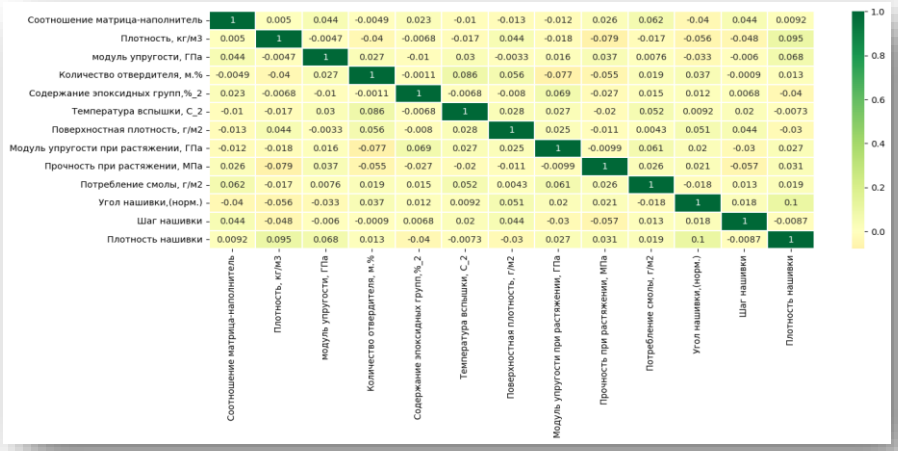
В анализе участвуют:

- MinMaxScaler()
- Normalizer()
- RobustScaler()

StandardScaler() не будем использовать, так как некоторые данные нет нормального распределения, так что не будем его использовать, будем исполь-зовать другие методы предпроцессинга.

Предпроцессинг

MinMaxScaler()



RobustScaler()

Normalizer()

Метрики качества

- R2 или коэффициент детерминации
- MAE (Mean Absolute Error) или средняя абсолютная ошибка

Модели и Метрики качества

- Линейная регрессия
- Метод k-ближайших соседей
- Деревья решений
- Случайный лес
- Нейронная сеть

- R2 или коэффициент детерминации
- MAE (Mean Absolute Error) или средняя абсолютная ошибка

Результаты построения и обучения моделей

	Model	MAE	R2 score
Прочность при растяжении	RandomForestRegressor_pr	0.008056	0.963
Прочность при растяжении	KNeighborsRegressor_pr	0.008650	0.962
Прочность при растяжении	LinearRegression_pr	0.011843	0.950
Прочность при растяжении	DecisionTreeRegressor_pr	0.015367	0.895
Модуль упругости при растяжении	RandomForestRegressor_upr	0.00097	0.787
Модуль упругости при растяжении	LinearRegression_upr	0.00099	0.787
Модуль упругости при растяжении	DecisionTreeRegressor_upr	0.00108	0.787
Модуль упругости при растяжении	KNeighborsRegressor_upr	0.001122	0.729

Модель для соотношения матрица-наполнитель

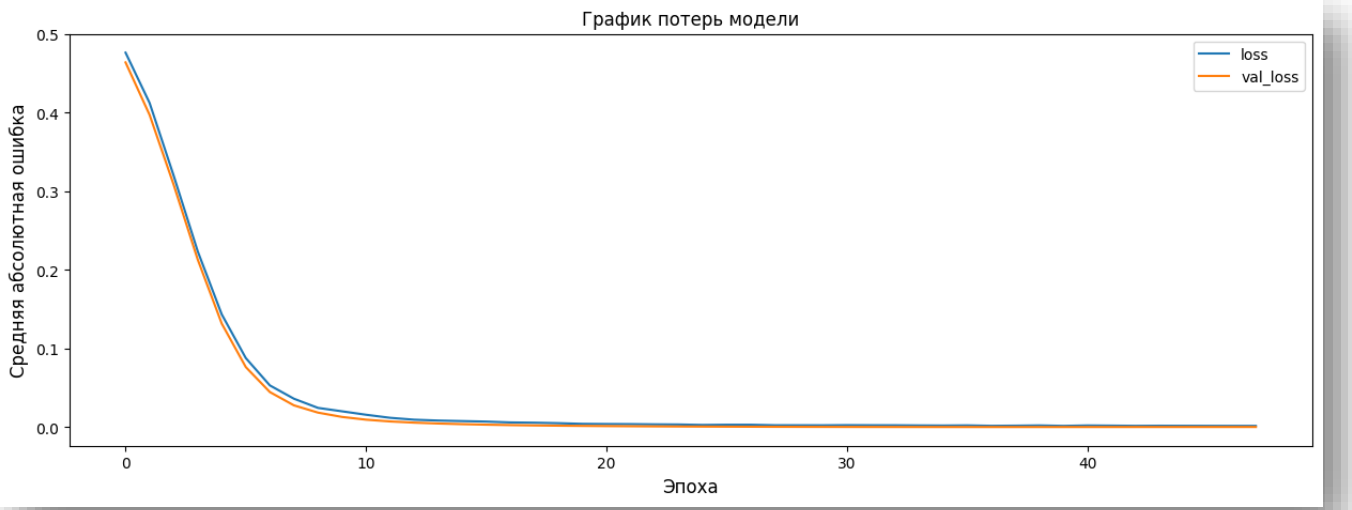
```
#архитектура модели

model_ns = tf.keras.Sequential()

model_ns.add(Dense(16, input_dim=X_train.shape[1], activation = 'relu'))
model_ns.add(BatchNormalization())
model_ns.add(Dense(8, activation = 'relu'))
model_ns.add(Dropout(0.18))
model_ns.add(Dense(8, activation = 'relu'))
model_ns.add(Dense(1, activation = 'sigmoid'))
```

Архитектура модели

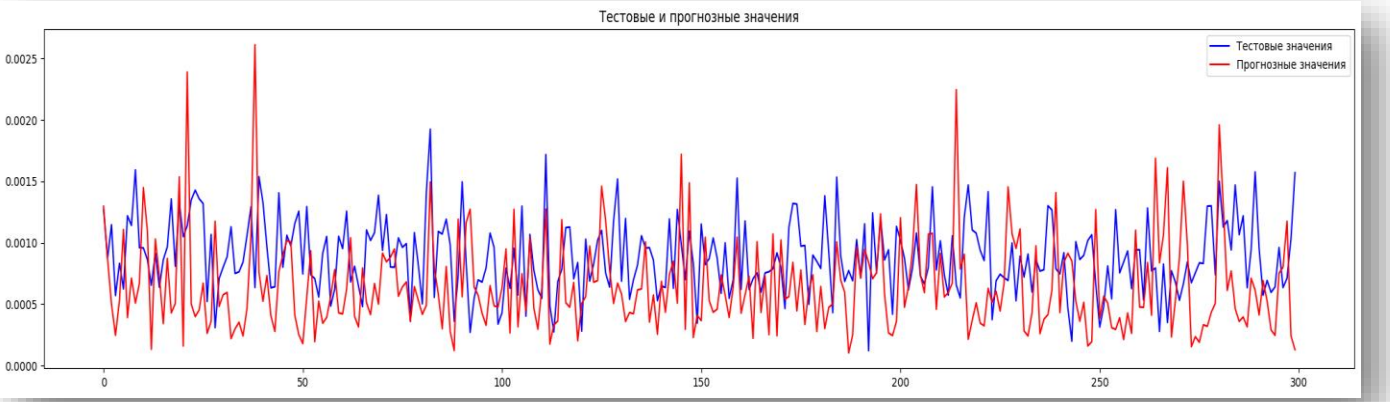
Построение нейросети

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense (Dense)               (None, 16)                208

 batch_normalization (Batch  (None, 16)                64
 Normalization)

 dense_1 (Dense)             (None, 8)                 136

 dropout (Dropout)           (None, 8)                 0

 dense_2 (Dense)             (None, 8)                 72

 dense_3 (Dense)             (None, 1)                 9

=================================================================
Total params: 489 (1.91 KB)
Trainable params: 457 (1.79 KB)
Non-trainable params: 32 (128.00 Byte)
_________________________________________________________________
```

Модель для соотношения матрица-наполнитель



Построение График потерь



Построение График потерь

Модель для соотношения матрица-наполнитель

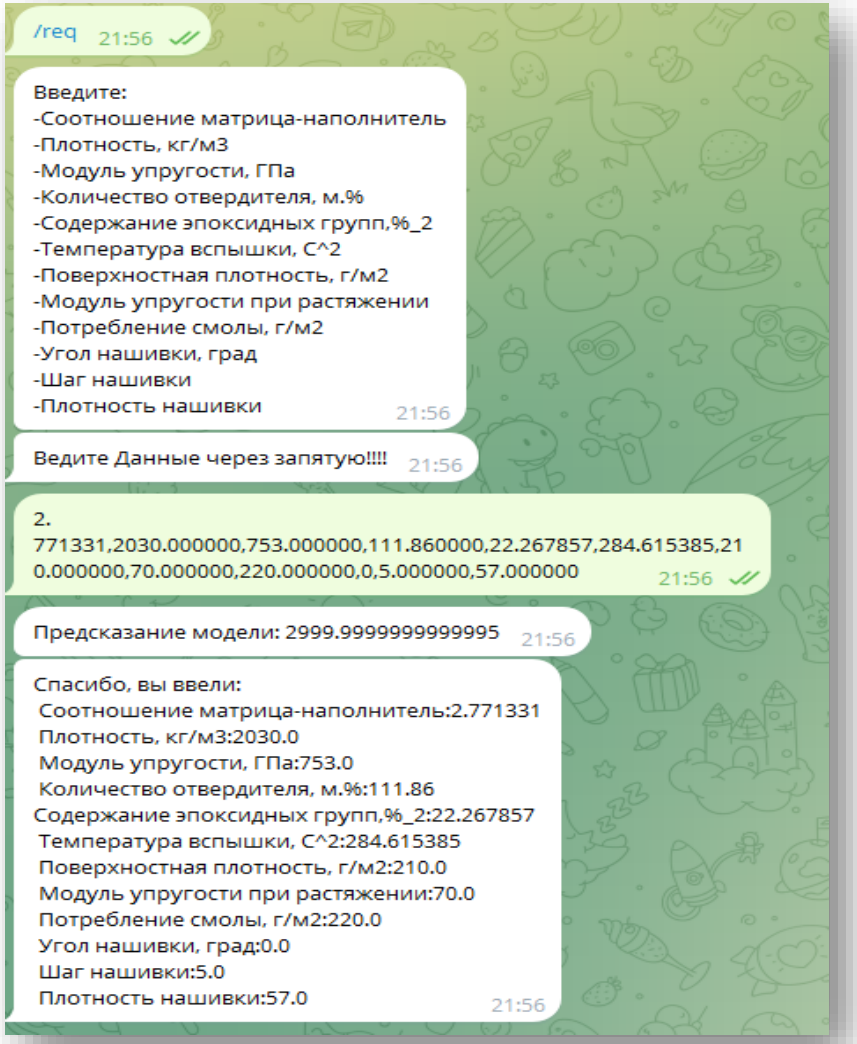
График прогнозных и настоящих значений

Разработка Чат-бота

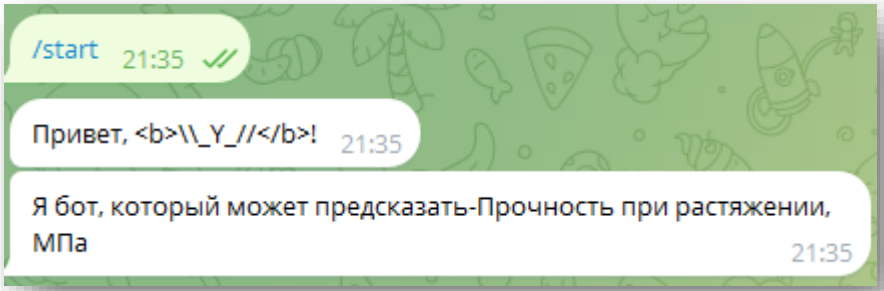
Получаем название нашего, бота, внутренние команды

Библиотека по разработке чат-бота

Разработка Чат-бота



Команда /req



Команда /start

- На выходе пользователь получает результат прогноза для значения параметра «Прочность при растяжении, МПа».

Результаты

Задача Решена

Выводы:

- Выявление причин отсутствие корреляции между признаками
- Модели машинного обучения прогнозируют точный результат, с незначительной ошибкой
- Нейросеть прогнозирует точный результат, с незначительной ошибкой
- Разработка чат-бота

Спасибо за внимание!