



ЦЕНТР  
ДОПОЛНИТЕЛЬНОГО  
ОБРАЗОВАНИЯ  
МГТУ им. Н.Э. Баумана

# Выпускная квалификационная работа по курсу «Data Science»

Слушатель Андреева Елена Анатольевна

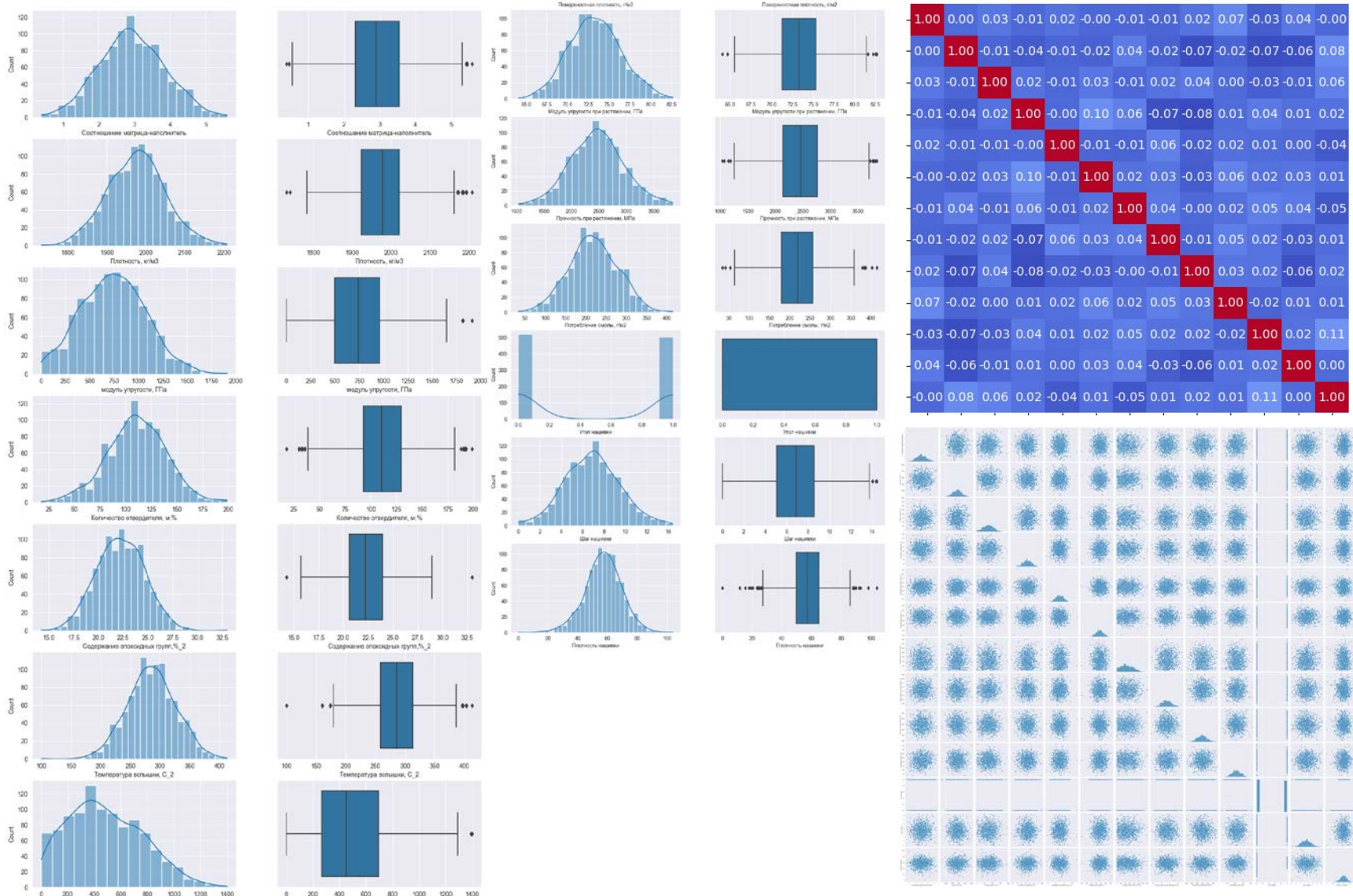
## Этапы выполнения работы

- ❖ Разведочный анализ данных
- ❖ Предобработка данных
- ❖ Разработка и обучение регрессионных моделей для прогнозирования «Модуль упругости при растяжении, ГПа» и «Прочность при растяжении, МПа»
- ❖ Построение нейронной сети для рекомендации «Соотношение матрица-наполнитель»



ЦЕНТР  
ДОПОЛНИТЕЛЬНОГО  
ОБРАЗОВАНИЯ  
МГТУ им. Н.Э. Баумана

# Разведочный анализ данных



- описательная статистика данного датасета;
- визуальный анализ гистограмм;
- визуальный анализ диаграмм размаха («ящик с усами»);
- проверка нормальности распределения по критерию Пирсона;
- анализ попарных графиков рассеяния переменных;
- корреляционный анализ с целью поиска коэффициентов



# Предобработка данных

Найдем выбросы.

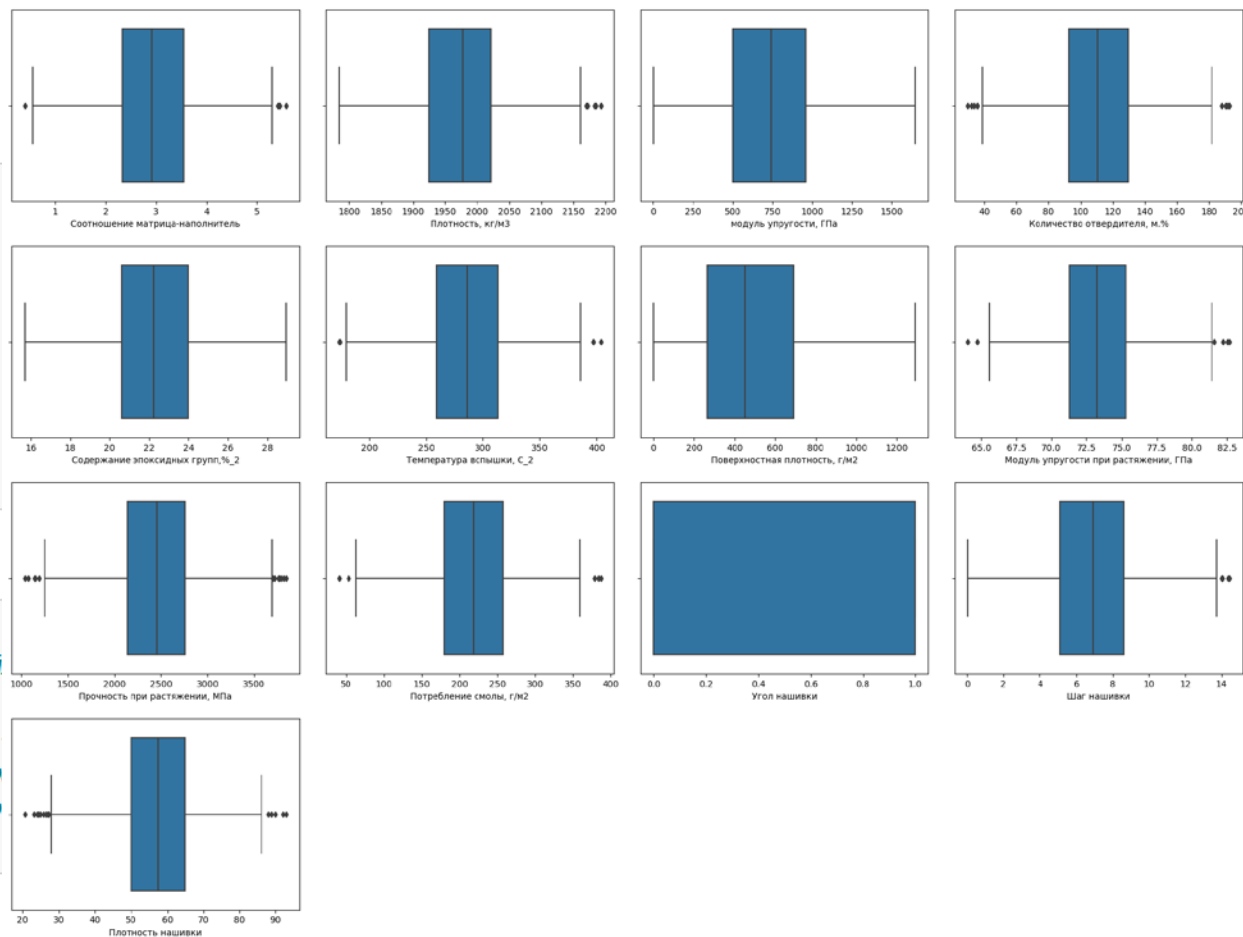
Для поиска выбросов воспользуемся методом 3-х сигм.

```
count_3s = 0
count_iq = 0
for column in df1:
    d = df1.loc[:, [column]]
    # методом 3-х сигм
    zscore = (df1[column] - df1[column].mean()) / df1[column].std()
    d['3s'] = zscore.abs() > 3
    count_3s += d['3s'].sum()
print('Метод 3-х сигм, выбросов:', count_3s)
```

Метод 3-х сигм, выбросов: 24

```
# Удаляем выбросы в датасете с помощью метода трёх сигм
outliers = pd.DataFrame(index=df1.index) # Создание пустого датафрейма
for column in df1: # запускаем цикл по каждому столбцу датафрейма
    zscore = (df1[column] - df1[column].mean()) / df1[column].std()
    outliers[column] = (zscore.abs() > 3) # определяем выбросы с помо
df1 = df1[outliers.sum(axis=1)==0] # фильтруем, оставляя только стро
df1.shape
```

(1000, 13)





## Обучение моделей

Для прогнозирования модуля упругости при растяжении и прочности при растяжении были использованы следующие методы машинного обучения:

Модуля упругости при растяжении, ГПа

	RMSE	MAE	R2
<b>Lasso</b>	-0.021946	3.104460	2.512111
<b>LinearRegression</b>	-0.042239	3.135130	2.542403
<b>Ridge</b>	-0.036349	3.126259	2.533514
<b>DecisionTreeRegressor</b>	-0.065842	3.170432	2.551608
<b>RandomForestRegressor</b>	-0.058812	3.159959	2.527859
<b>ElasticNetCV</b>	-0.021981	3.104512	2.510919
<b>SVR</b>	-0.062883	3.166028	2.556054
<b>BayesianRidge</b>	-0.021522	3.103815	2.510063
<b>KernelRidge</b>	-0.031442	3.118849	2.524650

Модель для расчёта прочности при растяжении, МПа

	R2	RMSE	MAE
<b>Lasso</b>	-0.019856	469.004096	365.853564
<b>LinearRegression</b>	-0.041048	473.851843	370.003339
<b>Ridge</b>	-0.034646	472.392702	368.940470
<b>DecisionTreeRegressor</b>	-0.025416	470.280839	369.984701
<b>RandomForestRegressor</b>	-0.037358	473.011207	372.443349
<b>ElasticNetCV</b>	-0.000928	464.631542	363.886617
<b>SVR</b>	-0.049553	475.783433	368.535000
<b>BayesianRidge</b>	-0.000928	464.631544	363.886619
<b>KernelRidge</b>	-0.010346	466.812301	364.936087



# Нейронная сеть для рекомендации «Соотношение матрица-наполнитель»

Результаты работы нейронных сетей для предсказания “Соотношение матрица-наполнитель”

	<b>R2</b>	<b>RMSE</b>
<b>MLPRegressor без подобранных гиперпараметров</b>	-0.037495	0.940855
<b>MLPRegressor с подобранными гиперпараметрами</b>	-0.019444	0.932634
<b>Последовательная нейросеть (Keras)</b>	-0.064893	0.953197
<b>Последовательная нейросеть (Keras) с callback</b>	-0.085778	0.962499
<b>Последовательная нейросеть (Keras) с Dropout</b>	-0.048021	0.945616



ЦЕНТР  
ДОПОЛНИТЕЛЬНОГО  
ОБРАЗОВАНИЯ  
МГТУ им. Н.Э. Баумана



[do.bmstu.ru](https://do.bmstu.ru)