



ЦЕНТР
ДОПОЛНИТЕЛЬНОГО
ОБРАЗОВАНИЯ
МГТУ им. Н.Э. Баумана

Выпускная квалификационная работа по курсу «Data Science»

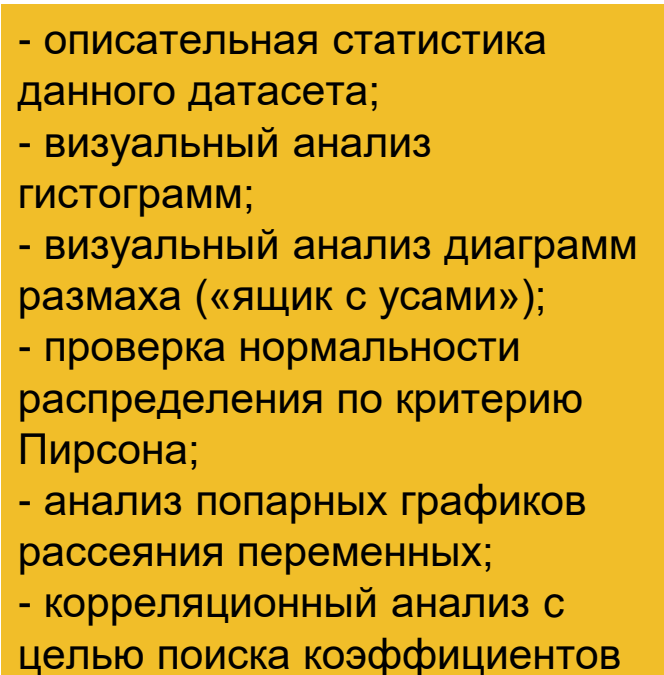
Слушатель Андреева Елена Анатольевна

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»
- ❖ Разработка веб - приложения Flask



]





Предобработка данных

Найдем выбросы.

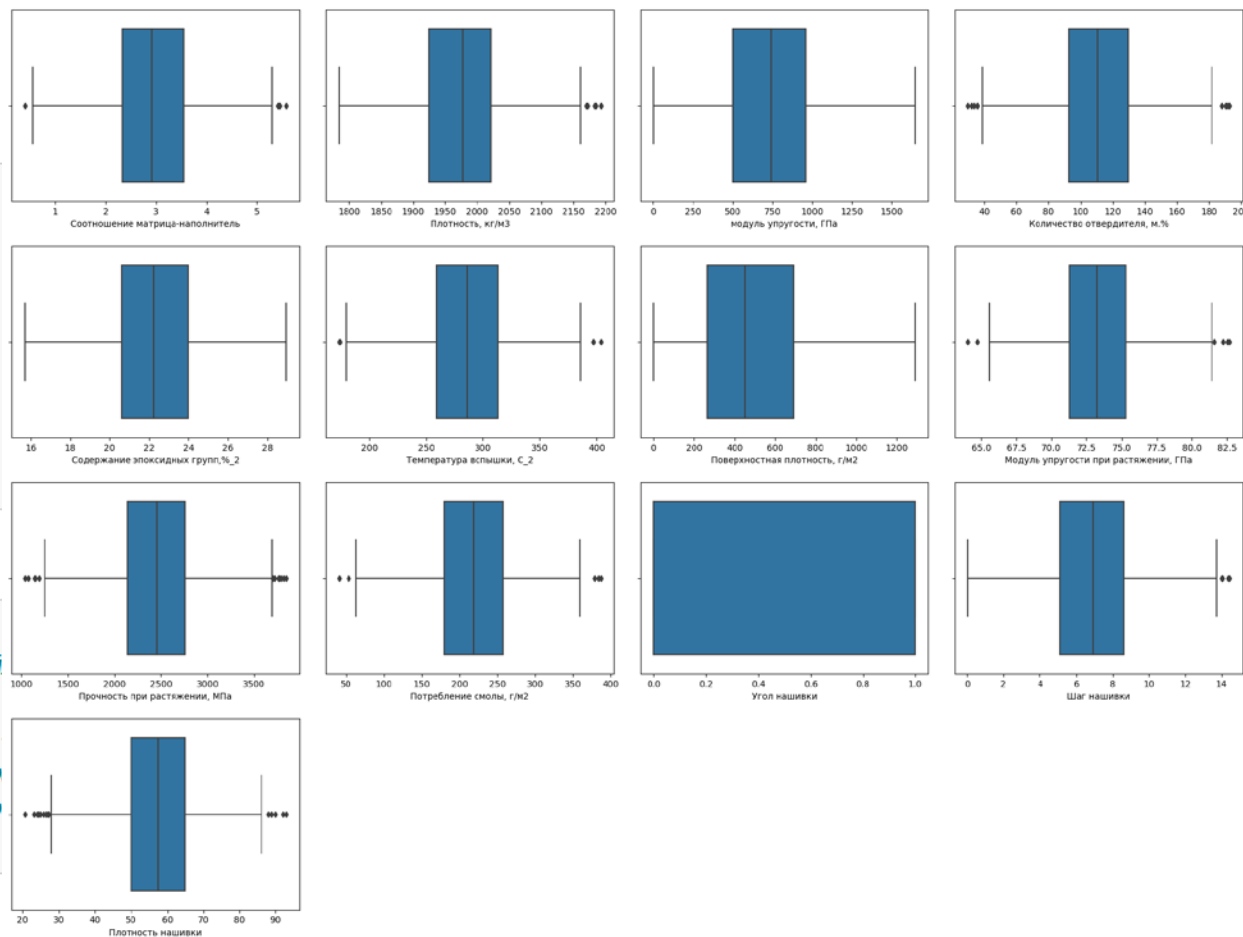
Для поиска выбросов воспользуемся методом 3-х сигм.

```
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)





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

- Лассо-регрессия (Lasso);
- Линейная регрессия (LinearRegression);
- Гребневая регрессия (Ridge);
- Регрессионное дерево решений (DecisionTreeRegressor);
- Случайный лес регрессии (RandomForestRegressor);
- Эластичная сеть регрессии (ElasticNetCV);
- Метод опорных векторов для регрессии (SVR);
- Байесовская линейная регрессия (BayesianRidge);
- Ядерная регрессия (KernelRidge).

Нейронная сеть для рекомендации «Соотношение матрица-наполнитель»

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

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



ЦЕНТР
ДОПОЛНИТЕЛЬНОГО
ОБРАЗОВАНИЯ
МГТУ им. Н.Э. Баумана



do.bmstu.ru